



Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

1969-06

Nonsimilar solution of the laminar boundary layer in an oscillatory flow by an integral matrix method

Gastrock, Barry Allen

Monterey, California. U.S. Naval Postgraduate School

<http://hdl.handle.net/10945/12729>

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>

NPS ARCHIVE
1969.06
GASTROCK, B.

NONSIMILAR SOLUTION OF THE LAMINAR
BOUNDARY LAYER IN AN OSCILLATORY
FLOW BY AN INTEGRAL MATRIX METHOD

by

Barry Allen Gastrock

United States Naval Postgraduate School



THESIS

NONSIMILAR SOLUTION OF THE LAMINAR
BOUNDARY LAYER IN AN OSCILLATORY
FLOW BY AN INTEGRAL MATRIX METHOD

by

Barry Allen Gastrock

June 1969

*This document has been approved for public re-
lease and sale; its distribution is unlimited.*

LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93940

Nonsimilar Solution of the Laminar Boundary Layer
in an Oscillatory Flow by an Integral Matrix Method

by

Barry Allen Gastrock
Lieutenant, United States Navy
B.S., United States Naval Academy, 1961

Submitted in partial fulfillment of the
requirements for the degree of

AERONAUTICAL ENGINEER

from the
NAVAL POSTGRADUATE SCHOOL
June 1969

ABSTRACT

The development of a numerical procedure for the treatment of nonsimilar, unsteady, laminar boundary layers is presented. The solution is obtained from the laminar, isothermal, incompressible boundary layer equations employing a modification of the integral matrix procedure of Bartlett and Kendall. Solutions of example problems are presented for steady Blasius and Howarth flows, and for oscillating Blasius flow. Agreement with the known classical results is satisfactory and establishes the general feasibility of the method. Core storage requirements of 130,000 bytes allow consideration of as many as 25 nodal points across the boundary layer, 50 time increments per oscillation cycle and 50 streamwise stations. Solution of oscillating Blasius flow considering 8 nodal points and 16 time increments requires 13.49 seconds for one streamwise station utilizing IBM 360/67 time sharing capabilities.

TABLE OF CONTENTS

I.	INTRODUCTION -----	13
A.	GENERAL -----	13
B.	ANALYTICAL HISTORY -----	13
C.	APPLICATION OF NUMERICAL METHODS -----	14
II.	ANALYSIS -----	16
A.	GOVERNING EQUATIONS -----	16
B.	BOUNDARY CONDITIONS -----	17
C.	TRANSFORMATION OF EQUATIONS -----	18
D.	TAYLOR SERIES EXPANSIONS -----	20
E.	AXIAL AND TIME DERIVATIVES -----	21
F.	INTEGRATION OF THE BOUNDARY LAYER EQUATION ---	23
G.	RECURRENCE FORMULAS -----	24
H.	MATRIX FORMULATION -----	28
I.	SOLUTION PROCEDURE -----	32
III.	RESULTS AND DISCUSSION -----	34
A.	STEADY BLASIOUS FLOW -----	34
B.	STEADY HOWARTH FLOW -----	34
C.	OSCILLATING BLASIOUS FLOW -----	37
D.	NUMERICAL PROCEDURE -----	37
IV.	CONCLUSIONS AND RECOMMENDATIONS -----	40
APPENDIX A	TAYLOR SERIES EXPANSIONS OF INTEGRALS -----	41
APPENDIX B	BOUNDARY LAYER EQUATION CORRECTION COEFFICIENTS -----	43
APPENDIX C	PROGRAM DESCRIPTION -----	44
APPENDIX D	INPUT INSTRUCTIONS FOR THE UBLIMP PROGRAM -----	52

SAMPLE COMPUTER PROBLEM -----	60
COMPUTER PROGRAM -----	70
LIST OF REFERENCES -----	96
INITIAL DISTRIBUTION LIST -----	98
FORM DD 1473 -----	99

LIST OF FIGURES

1. Linear Portion of $[A]$ Matrix -----	30
2. Nonlinear Portion of $[A]$ Matrix -----	31
3. Steady Blasius Flow Data Comparison -----	35
4. Steady Howarth Flow Data Comparison -----	36
5. Oscillating Blasius Flow Data Comparison -----	38
6. Computer Program Overlay Structure -----	45
7. Schematic of UBLIMP Subroutines -----	46
8. Functional Flow Chart of UBLIMP Program -----	47

TABLE OF SYMBOLS

b_1	Rate of change of the mean free stream velocity with distance from the leading edge
d_0, d_1, d_2	Coefficients defined in the finite difference representation of streamwise derivatives (defined in equations (33) and (34) for two and three-point difference relations)
f	Stream function (defined by equation (14))
P	Pressure
r_0	Local radius of the body in a meridian plane for an axisymmetric shape
s	Distance along the body from a stagnation point or the leading edge
T_0, T_1, T_2	Coefficients defined in the finite difference representation of time derivatives (defined in equations (37) and (38) for two and three-point difference relations)
t	Time
U_0	Mean free stream velocity at the leading edge
U	s velocity component outside the boundary layer
u	Velocity component parallel to the body surface
v	Velocity component normal to the body surface
$XM_1, XM_3, \dots$	Truncated series obtained in Taylor series expansion of $\int_{i-1}^i f'^2 d\eta$ (defined by equation (A2))
y	Distance from the surface into the boundary layer measured normal to the surface
$ZM_1, ZM_2, \dots$	Truncated series obtained in Taylor series expansion of integrals involving nonsimilar terms (defined by equation (A7))

α^*	Normalizing parameter (defined by equation (16))
β	Streamwise pressure gradient parameter (defined by equation (22))
k^{Δ}_{k-1}	Time difference between two time grid points denoted by the subscripts k and $k-1$ (defined by equation (39))
$\ell^{\Delta}_{\ell-1}$	Logarithmic distance between two streamwise positions denoted by the subscripts ℓ and $\ell-1$ (defined by equation (35))
$\Delta f_i, \Delta f'_i, \dots$	Corrections for $f_i, f'_i, \dots$ during Newton-Raphson iterations
$\delta\eta$	Distance between two boundary layer nodal points (defined by equation (30))
δt	Time difference between two time grid points (defined by equation (42))
ϵ	Ratio of the amplitude of the free stream oscillation to the local mean free stream velocity
η	Transformed coordinate in a direction normal to the surface (defined by equation (10))
μ	Shear viscosity
ξ	Transformed streamwise coordinate (defined by equation (9))
ρ	Density
ω	Frequency of the free stream oscillation

SUBSCRIPTS

e	Pertains to the boundary layer edge
i	Pertains to the i^{th} nodal point in the boundary layer, starting with $i=1$ at the surface
k	Pertains to the k^{th} time grid in the oscillation cycle
ℓ	Pertains to the ℓ^{th} streamwise position

m Pertains to the m^{th} iteration during the
Newton-Raphson iteration

w Pertains to the wall or surface of the body

SUPERSCRIPTS

n Equal to unity for axisymmetric bodies and
zero for two-dimensional bodies

' (prime) Represents partial differentiation with respect
to η

MATRIX NOTATION

$[A]$ Coefficients of corrections matrix in Newton-
Raphson recurrence formulas prior to any matrix
reduction (defined by equation (60))

$[AL]$ Coefficients of "linear" corrections in Newton-
Raphson recurrence formulas corresponding to
linear equations prior to any matrix reduction
(defined by equation (61))

$[ANL]$ Coefficients of "linear" corrections in Newton-
Raphson recurrence formulas corresponding to
nonlinear equations prior to any matrix reduc-
tion (defined by equation (61))

$[BAL]$ Product of $[AL]^{-1}[BL]$ (defined in equation (66))

$[BL]$ Coefficients of "nonlinear" corrections in
Newton-Raphson recurrence formulas correspond-
ing to linear equations prior to any matrix
reduction (defined by equation (61))

$[BNL]$ Coefficients of "nonlinear" corrections in
Newton-Raphson recurrence formulas correspond-
ing to nonlinear equations prior to any matrix
reduction (defined by equation (61))

$[\overline{BNL}]$ Coefficients of "nonlinear" corrections in
Newton-Raphson recurrence formulas correspond-
ing to nonlinear equations after matrix reduc-
tion (defined in equation (69))

$[E]$ Matrix of errors in Newton-Raphson recurrence
formulas prior to any matrix reduction (defined
in equation (60))

$[EL]$ Matrix of errors in Newton-Raphson recurrence
formulas corresponding to linear equations prior
to any matrix reduction (defined by equation
(61))

$[ELA]$	Product of $[AL]^{-1}[-EL]$ (defined in equation (67))
$[ENL]$	Matrix of errors in Newton-Raphson recurrence formulas corresponding to nonlinear equations prior to any matrix reduction (defined by equation (61))
$[\overline{ENL}]$	Matrix of errors in Newton-Raphson recurrence formulas corresponding to nonlinear equations after matrix reduction (defined by equation (70))
$[\Delta V]$	Matrix of corrections to primary variables in Newton-Raphson recurrence formulas (defined by equation (60))
$[\Delta VL]$	Matrix of corrections to "linear" primary variables in Newton-Raphson recurrence formulas (defined by equation (61))
$[\Delta VNL]$	Matrix of corrections to "nonlinear" primary variables in Newton-Raphson recurrence formulas (defined by equation (61))

ACKNOWLEDGEMENT

The author wishes to express his particular appreciation to Dr. Robert M. Kendall for his assistance, inspiration and interest in the problem. The practical experience gained through working with Dr. Kendall has proven invaluable in the conduct of this investigation.

I. INTRODUCTION

A. GENERAL

All aerodynamic flows are to some degree influenced by unsteadiness. Rotating stall in turbomachinery, aerodynamic stall flutter and hydrofoil flow fields are but a few practical problems in which the nonsteady fluid flow phenomena are of prime importance. Additionally, the recent emphasis on the development of rotary wing aircraft has created a new demand for knowledge of the behavior of boundary layers in unsteady flow.

B. ANALYTICAL HISTORY

Early analytical treatments by Stokes, Rayleigh and Schlichting (Ref. 1) considered only a special case of the general problem: that of unsteady viscous fluid flow with no mean flow or pressure gradient. Lighthill (Ref. 2) made the first significant investigation of small sinusoidal oscillations superimposed on a mean flow. Lighthill's analysis was based on a small perturbation treatment of the boundary layer equations for sinusoidal flow, with only the first order terms retained. The determination of the existence of the "quasi-steady" regime at low frequencies, and the "shear wave" regime at higher frequencies was the principal result of this analysis. Glauert (Ref. 3) extended Lighthill's work by considering the boundary layer in the vicinity of a stagnation point. Carrier and DiPrima (Ref. 4),

using a system of equations derived using a modified Oseen linearization, confirmed the existence of the "shear wave" solution and the phase advance of the wall shear determined by Lighthill. Nickerson (Ref. 5) initially retained perturbation terms beyond the first order, thus widening the scope of Lighthill's work.

In addition to his analytical development, Nickerson made hot-wire measurements of the laminar boundary layer on a flat plate oscillating in a Blasius mean flow. His work partially confirmed both his own and Lighthill's analyses, but mechanical difficulties precluded the completion of the experimental program. Hill and Stenning (Ref. 6) imposed sinusoidal oscillations directly on mean Blasius and Howarth flows and were able to confirm the analytical results of both Lighthill and Nickerson. Recently an integral analysis of the unsteady boundary layer was introduced by Koob and Abbott (Ref. 7), who succeeded in demonstrating the method for the case of an accelerating flat plate.

C. APPLICATION OF NUMERICAL METHODS

With the advent of the high-speed digital computer, it has become possible to apply numerical methods to the analytical studies of viscous fluid flows. Fromm (Ref. 8) and Chorin (Ref. 9) introduced modified finite difference methods for the treatment of viscous flows which could feasibly be applied to the boundary layer problem. Smith and Clutter (Ref. 10) and Kleinstein and Nabi (Refs. 11 and

12) applied finite difference methods to the steady boundary layer equations. Successive approximation, linearization and integral methods have also been applied to the boundary layer equations (Refs. 13, 14 and 15). A major drawback of most of the methods mentioned is that in order to reach a solution to a given problem large computing times are necessary.

The introduction of the Boundary Layer Integral Matrix Procedure by Bartlett and Kendall (Ref. 16) provided a major breakthrough in the field of fast computer solution of the steady flow boundary layer problem. The method was developed as a means of obtaining rapid computations in the laminar regime for steady, multicomponent, chemically reacting boundary layers; and the logical extension of the technique was to encompass the unsteady flow condition.

The purpose of this investigation is to develop a computer program capable of solving the unsteady, laminar boundary layer problem by adapting the rapid solution techniques of the Boundary Layer Integral Matrix Procedure.

II. ANALYSIS

A. GOVERNING EQUATIONS

In this development, utilizing the integral matrix method, it is convenient to make the assumption of incompressible isothermal flow. The resulting laminar boundary layer equations for two-dimensional ($n=0$) or axisymmetric ($n=1$) flow are as follows:

Continuity:

$$\frac{1}{r_o^n} \frac{\partial (r_o^n u)}{\partial s} + \frac{\partial v}{\partial y} = 0 \quad (1)$$

Momentum:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial s} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial P}{\partial s} + \frac{\mu}{\rho} \frac{\partial^2 u}{\partial y^2} \quad (2)$$

$$\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial s} = -\frac{1}{\rho} \frac{\partial P}{\partial s} \quad (3)$$

where s and y are respectively the streamwise and normal components, u and v are the corresponding velocity components, U is the streamwise velocity component outside the boundary layer, r_o is the radius of the body perpendicular to the axis of revolution for an axisymmetric body, n is zero for a two-dimensional body and unity for an axisymmetric body, t is time, ρ is density, μ is viscosity and P is pressure.

B. BOUNDARY CONDITIONS

The boundary conditions considered for the governing equations as stated are:

$$u_w = 0 \quad (4)$$

$$v_w = 0 \quad (5)$$

$$u_e = U(s, t) \quad (6)$$

$$\frac{\partial u_e}{\partial y} = 0 \quad (7)$$

where the subscripts w and e refer respectively to wall and boundary layer edge conditions. Equation (4) is the requirement of zero slip at the wall, equation (5) implies no mass transfer at the wall, and equations (6) and (7) are representative of the assumption that the boundary layer edge velocity is equal to the free stream velocity.

The free stream velocity may be written:

$$U = (U_0 - b_1 s) (1 + \epsilon \cos \omega t) \quad (8)$$

where U_0 is the mean free stream velocity at the leading edge, b_1 is the rate of change of the mean free stream velocity with distance from the leading edge, ϵ is the ratio of the amplitude of the free stream oscillation to the local mean free stream velocity and ω is the frequency of the oscillation.

C. TRANSFORMATION OF EQUATIONS

Paralleling the development of Bartlett and Kendall (Ref. 16) a transformation of variables is performed utilizing a combination of the Levy and Mangler and the Howarth-Dorodnitsyn transformations which is known by a variety of names including Levy-Lees, Lees-Dorodnitsyn and Mangler-Dorodnitsyn. This transformation is as follows:

$$\xi = \int_0^s u_e \rho_e u_e r_o^{2n} ds \quad (9)$$

$$\eta = \frac{r_o^n u_e}{\sqrt{2\xi}} \int_0^y \rho dy \quad (10)$$

Simplifying equations (9) and (10) to encompass the assumptions of incompressibility and constant temperature the transformations become:

$$\xi = \rho \mu \int_0^s u_e r_o^{2n} ds \quad (11)$$

$$\eta = \frac{r_o^n \rho u_e}{\sqrt{2\xi}} y \quad (12)$$

The transformation to Levy-Lees variables is greatly simplified by use of the operator:

$$u \frac{\partial ()}{\partial s} + v \frac{\partial ()}{\partial s} = \alpha^* u_e r_o^n \sqrt{2\xi} \left\{ \epsilon' \frac{\partial ()}{\partial \xi} - \left(\frac{f}{2\xi} + \frac{\partial f}{\partial \xi} \right) \frac{\partial ()}{\partial \eta} \right\} \quad (13)$$

where f is the stream function defined as:

$$f - f_w = \int_0^\eta \frac{u}{u_e} d\eta \quad (14)$$

the prime represents partial differentiation with respect to η so that:

$$f' = \frac{u}{u_e} \quad (15)$$

and α^* is a normalizing parameter:

$$\alpha^* = \frac{\rho \mu u_e r_o^n}{\sqrt{2\xi}} \quad (16)$$

In addition, the partial derivatives are:

$$\frac{\partial(\quad)}{\partial s} = \alpha^* r_o^n \sqrt{2\xi} \left\{ \frac{\partial(\quad)}{\partial \xi} + \frac{\partial \eta}{\partial \xi} \frac{\partial(\quad)}{\partial \eta} \right\} \quad (17)$$

$$\frac{\partial(\quad)}{\partial y} = \frac{\alpha^*}{\mu} (\quad)' \quad (18)$$

$$\frac{\partial^2(\quad)}{\partial y^2} = \left(\frac{\alpha^*}{\mu} \right)^2 (\quad)'' \quad (19)$$

Applying equation (15) to the velocity in the partial derivative with respect to time yields:

$$\frac{\partial u}{\partial t} = u_e \frac{\partial f'}{\partial t} + f' \frac{\partial u_e}{\partial t} \quad (20)$$

Using equations (13) through (20) and applying equation (6) to equation (3) results in the following transformed momentum equation:

$$f''' + ff'' + \beta(1 - f'^2) - 2 \left\{ f' \frac{\partial f'}{\partial \ln \xi} - f'' \frac{\partial f}{\partial \ln \xi} \right\} - \frac{\rho u}{\alpha^* 2} \left\{ \frac{\partial f'}{\partial t} + (f' - 1) \frac{\partial \ln u_e}{\partial t} \right\} = 0 \quad (21)$$

where

$$\beta = 2 \frac{\partial \ln u_e}{\partial \ln \xi} \quad (22)$$

Transforming the boundary conditions, equations (4) through (7) become:

$$f'_w = 0 \quad (23)$$

$$f_w = 0 \quad (24)$$

$$f'_e = 1 \quad (25)$$

$$f''_e = 0 \quad (26)$$

D. TAYLOR SERIES EXPANSIONS

At any streamwise station s , the boundary layer may be divided into $N-1$ strips which are joined by N nodal points η_i , where i is 1 at the wall and N at the boundary layer edge. The primary dependent variables f_i and their normal derivatives are related by Taylor series expansions such that the f'_i are represented by connected cubics with continuous first and second derivatives at the nodal points. This is commonly called a spline fit.

Expanding the primary variables in Taylor series and truncating at f'''_i the following linear equations result:

$$-f_{i+1} + f_i + f_i' \delta\eta + f_i'' \frac{\delta\eta^2}{2} + f_i''' \frac{\delta\eta^3}{8} + f_{i+1}''' \frac{\delta\eta^3}{24} = 0 \quad (27)$$

$$-f_{i+1}' + f_i' + f_i'' \delta\eta + f_i''' \frac{\delta\eta^3}{3} + f_{i+1}''' \frac{\delta\eta^2}{6} = 0 \quad (28)$$

$$-f_{i+1}'' + f_i'' + f_i''' \frac{\delta\eta}{2} + f_{i+1}''' \frac{\delta\eta}{2} = 0 \quad (29)$$

$$\delta\eta = \eta_{i+1} - \eta_i \quad (30)$$

$$f_{i+1}''' = \frac{f_{i+1}''' - f_i'''}{\delta\eta} \quad (31)$$

Equations (27) through (29) represent a set of 3(N-1) equations where f is represented as a quartic, f' as a cubic, f'' as a quadratic and f''' is considered to vary linearly between η_i and η_{i+1} .

E. AXIAL AND TIME DERIVATIVES

The axial derivatives of equation (21) are treated as logarithmic two or three-point differences such that:

$$2 \left[\frac{\partial(\)}{\partial \ln \xi} \right]_{\ell} = d_0(\)_{\ell} + d_1(\)_{\ell-1} + d_2(\)_{\ell-2} \quad (32)$$

where for two-point differences:

$$d_0 = \frac{2}{\ell \Delta_{\ell-1}} \quad d_1 = -d_0 \quad d_2 = 0 \quad (33)$$

and for three-point differences:

$$d_0 = 2 \frac{\ell^{\Delta} \ell-1 + \ell^{\Delta} \ell-2}{\ell^{\Delta} \ell-1 \ell^{\Delta} \ell-2} \quad d_1 = -2 \frac{\ell^{\Delta} \ell-2}{\ell^{\Delta} \ell-1 \ell-1 \ell^{\Delta} \ell-2} \quad d_2 = 2 \frac{\ell^{\Delta} \ell-1}{\ell^{\Delta} \ell-2 \ell-1 \ell^{\Delta} \ell-2} \quad (34)$$

In equations (32) through (34), () _{$\ell-1$} implies the previous streamwise station.

$$\ell^{\Delta} \ell-1 = \ln \xi_{\ell} - \ln \xi_{\ell-1} = \ln (\xi_{\ell} / \xi_{\ell-1}) \quad (35)$$

Differencing the time derivatives in the same way:

$$\left[\frac{\partial ()}{\partial t} \right]_k = T_0 ()_k + T_1 ()_{k-1} + T_2 ()_{k-2} \quad (36)$$

where for two-point differences:

$$T_0 = \frac{1}{k^{\Delta} k-1} \quad T_1 = -T_0 \quad T_2 = 0 \quad (37)$$

and for three-point differences:

$$T_0 = \frac{k^{\Delta} k-1 + k^{\Delta} k-2}{k^{\Delta} k-1 k^{\Delta} k-2} \quad T_1 = -\frac{k^{\Delta} k-2}{k^{\Delta} k-1 k-1 k^{\Delta} k-2} \quad T_2 = \frac{k^{\Delta} k-1}{k^{\Delta} k-2 k-1 k^{\Delta} k-2} \quad (38)$$

In equations (36) through (38), () _{$k-1$} implies the previous time.

$$k^{\Delta} k-1 = t_k - t_{k-1} \quad (39)$$

Equations (37) through (39) may be greatly simplified by considering the utilization of equal time spacings. The resulting equations for the treatment of time derivatives over equal time spacings are as follows:

Two point differences:

$$T_0 = \frac{1}{\delta t} \quad T_1 = -T_0 \quad T_2 = 0 \quad (40)$$

Three-point differences:

$$T_0 = \frac{3}{2\delta t} \quad T_1 = -\frac{2}{\delta t} \quad T_2 = \frac{1}{2\delta t} \quad (41)$$

where:

$$\delta t = t_k - t_{k-1} = t_{k-1} - t_{k-2} \quad (42)$$

F. INTEGRATION OF THE BOUNDARY LAYER EQUATION

The transformed streamwise momentum equation, equation (21), is integrated at constant ξ from η_{i-1} to η_i , yielding:

$$\begin{aligned} & \left[f'' \right]_{i-1}^i + \int_{i-1}^i f f'' d\eta + \beta \int_{i-1}^i (1 - f'^2) d\eta \\ & - 2 \int_{i-1}^i \left(f' \frac{\partial f'}{\partial \ln \xi} - f'' \frac{\partial f}{\partial \ln \xi} \right) d\eta - \frac{\rho \mu}{\alpha^* 2} \int_{i-1}^i \frac{\partial f'}{\partial t} d\eta \\ & - \frac{\rho \mu}{\alpha^* 2} \int_{i=1}^i (f' - 1) \frac{\partial \ln u_e}{\partial t} d\eta = 0 \end{aligned} \quad (43)$$

Expanding equation (43) in Taylor series and applying equation (36) to the time derivative terms:

$$\int_{i-1}^i \frac{\partial f'}{\partial t} d\eta = \left[\frac{\partial f}{\partial t} \right]_{i-1}^i$$

$$= [T_0 f_k + T_1 f_{k-1} + T_2 f_{k-2}] \quad (44)$$

$$\int_{i-1}^i (f' - 1) \frac{\partial \ln u_e}{\partial t} d\eta = \frac{\partial \ln u_e}{\partial t} \left\{ [f]_{i-1}^i - \delta\eta \right\}$$

$$= \left\{ [f]_{i-1}^i - \delta\eta \right\} [T_0 \ln u_{e_k} + T_1 \ln u_{e_{k-1}} + T_2 \ln u_{e_{k-2}}] \quad (45)$$

the integrated boundary layer equation becomes:

$$\left[f'' + f'(1 + d_0)f + (d_1 f_{k-1} + d_2 f_{k-2})f' - \frac{\rho u}{\alpha^* 2} \{ [T_0 f_k + T_1 f_{k-1} + T_2 f_{k-2}] \right.$$

$$\left. - f [T_0 \ln u_{e_k} + T_1 \ln u_{e_{k-1}} + T_2 \ln u_{e_{k-2}}] \right]_{i-1}^i$$

$$- (1 + \beta + 2d_0) [f'_i XM_1 + f''_i XM_2 + f'''_i XM_3 + f^{(4)}_i XM_4]$$

$$- 2 [f'_i ZM_1 + f''_i ZM_2 + f'''_i ZM_3 + f^{(4)}_i ZM_4]$$

$$+ \delta\eta \left\{ \beta + \frac{\rho u}{\alpha^* 2} [T_0 \ln u_{e_k} + T_1 \ln u_{e_{k-1}} + T_2 \ln u_{e_{k-2}}] \right\} = 0 \quad (46)$$

The Taylor series expansions, XM and ZM, of the integrals in equation (43) are listed in Appendix A.

G. RECURRENCE FORMULAS

The boundary layer equations, equation (46), and the Taylor series expansions of the primary variables, equations (27) through (29), together with the boundary conditions,

equations (23) through (26), are solved by Newton-Raphson iteration of 4N simultaneous equations in 4N primary variable unknowns. These are summarized as follows:

<u>Primary Variables</u>	<u>No. of Equations</u>
f	N
f'	N
f''	N
f'''	N
f	N
No. of Eqns. Req:	<u>4N</u>

	<u>Equation Nos.</u>	<u>No. of Equations</u>
Taylor Series Expansions	(27) - (29)	3(N-1)
Boundary Layer Equations	(46)	N-1
Boundary Conditions	(34) - (26)	<u>4</u>
No. of Eqns. Available:		<u>4N</u>

The Newton-Raphson solution technique is illustrated by considering two simultaneous nonlinear equations:

$$F(x,y) = 0 \qquad G(x,y) = 0 \qquad (47)$$

which have the solution $x=X$ and $y=Y$. Defining x_m and y_m as the values of x and y for the m^{th} iteration, the desired solution $f(X,Y)$ is expressed in the Taylor series expansions:

$$\begin{aligned}
 0 = F(X,Y) &= F(x_m, y_m) + (X - x_m) \frac{\partial F(x_m, y_m)}{\partial x} \\
 &+ (Y - y_m) \frac{\partial F(x_m, y_m)}{\partial y} + \dots \\
 0 = G(X,Y) &= G(x_m, y_m) + (X - x_m) \frac{\partial G(x_m, y_m)}{\partial x} \\
 &+ (Y - y_m) \frac{\partial G(x_m, y_m)}{\partial y} + \dots \qquad (48)
 \end{aligned}$$

In the Newton-Raphson method (X,Y) is replaced by (x_{m+1},y_{m+1}) and the nonlinear terms in x and y appearing in equations (48) are ignored yielding the set of recurrence formulas:

$$\begin{aligned}\Delta x_m \frac{\partial F(x_m, y_m)}{\partial x} + \Delta y_m \frac{\partial F(x_m, y_m)}{\partial y} &= -F(x_m, y_m) \\ \Delta x_m \frac{\partial G(x_m, y_m)}{\partial x} + \Delta y_m \frac{\partial G(x_m, y_m)}{\partial y} &= -G(x_m, y_m)\end{aligned}\quad (49)$$

where:

$$\Delta x_m = x_{m+1} - x_m \quad \Delta y_m = y_{m+1} - y_m \quad (50)$$

The Δx_m and Δy_m are corrections to be added to x_m and y_m respectively yielding x_{m+1} and y_{m+1} . $F(x_m, y_m)$ and $G(x_m, y_m)$ are the values of the original functions $F(x, y)$ and $G(x, y)$ evaluated for the value of the variables during the m^{th} iteration. Noting that $F(x_m, y_m)$ and $G(x_m, y_m)$ approach zero as the corrections approach zero, it is appropriate to consider them as errors associated with the original equations (47).

Differentiating the Taylor series expansions, equations (27) through (29), with respect to the primary variables in accordance with the derivation of equation (49) yields the following recurrence formulas for the m^{th} iteration:

$$(-1)\Delta f_{i+1} + (1)\Delta f_i + (\delta\eta)\Delta f'_i + \left(\frac{\delta\eta^2}{2}\right)\Delta f''_i + \left(\frac{\delta\eta^3}{8}\right)\Delta f'''_i + \left(\frac{\delta\eta^3}{24}\right)\Delta f_{i+1}''' = -\text{ERROR} \quad (51)$$

$$(-1)\Delta f'_{i+1} + (1)\Delta f'_i + (\delta\eta)\Delta f''_i + \left(\frac{\delta\eta^2}{3}\right)\Delta f'''_i + \left(\frac{\delta\eta^2}{6}\right)\Delta f'''_{i+1} = -\text{ERROR} \quad (52)$$

$$(-1)\Delta f''_{i+1} + (1)\Delta f''_i + \left(\frac{\delta\eta}{2}\right)\Delta f'''_i + \left(\frac{\delta\eta}{2}\right)\Delta f'''_{i+1} = -\text{ERROR} \quad (53)$$

where Δf_i represents the correction for f_i , the coefficients of the corrections represent the partial derivatives of the Taylor series expansions with respect to the primary variables, and the ERRORS are equations (29) through (31) respectively, evaluated for the values of the variables during the iteration.

The recurrence formulas for the boundary layer equations are:

$$\begin{aligned} & \left[\Delta f'' + \Delta f' (1+d_0) f + f' (1+d_0) \Delta f + \Delta f' (d_1 f_{\ell-1} + d_2 f_{\ell-2}) \right. \\ & \left. - \frac{\rho u}{\alpha^2} \{ T_0 \Delta f - \Delta f (T_0 \ell n u_{e_k} + T_1 \ell n u_{e_{k-1}} + T_2 \ell n u_{e_{k-2}}) \} \right]_{i-1}^i \\ & - 2(1 + \beta + 2d_0) \{ \Delta f'_i X_{M_1} + \Delta f''_i X_{M_2} + \Delta f'''_i X_{M_3} + \Delta f'''_{i-1} X_{M_4} \} \\ & - 2 \{ \Delta f'_i Z_{M_1} + \Delta f''_i Z_{M_2} + \Delta f'''_i Z_{M_3} + \Delta f_{i-1} Z_{M_4} \} = -\text{ERROR} \quad (54) \end{aligned}$$

where ERROR is the left-hand-side of equation (46) evaluated for the values of the variables during the iteration.

Finally, the recurrence formulas for the boundary conditions, equations (23) through (26), are:

$$\Delta f'_w = -\text{ERROR} = - (f'_w)_m \quad (55)$$

$$\Delta f_w = -\text{ERROR} = - (f_w)_m \quad (56)$$

$$\Delta f_e' = - \text{ERROR} = 1 - (f_e')_m \quad (57)$$

$$\Delta f_e'' = - \text{ERROR} = - (f_e'')_m \quad (58)$$

Equation (54) may be further refined by collecting terms and writing the equation in terms of correction coefficients as follows:

$$\begin{aligned} C1\Delta f_i + C2\Delta f_{i-1} + C3\Delta f_i' + C4\Delta f_{i-1}' \\ + C5\Delta f_i'' + C6\Delta f_{i-1}'' + C7\Delta f_i''' + C8\Delta f_{i-1}''' = - \text{ERROR} \end{aligned} \quad (59)$$

The correction coefficients C1 through C8 are given in Appendix B.

H. MATRIX FORMULATION

The coefficients for the recurrence formulas, equations (51) through (53) and (55) through (59), evaluated for the m^{th} iteration form a square matrix $[A]$ with $4N$ rows and columns. The matrix equation is:

$$[A][\Delta V] = -[E] \quad (60)$$

where the column matrix $[\Delta V]$ is the matrix of corrections to the primary variables, and the column matrix $[E]$ is the matrix of errors. The matrix $[A]$ contains many zeros, and great savings in computer storage space and computation time may be realized through proper ordering of the equations and variables.

The equations are first divided into linear and nonlinear sets, and at the same time, the variables are classified as linear or nonlinear, resulting in the partitioned matrix equation:

$$\begin{bmatrix} \text{AL} & \text{BL} \\ \text{ANL} & \text{BNL} \end{bmatrix} \begin{bmatrix} \Delta\text{VL} \\ \Delta\text{VNL} \end{bmatrix} = - \begin{bmatrix} \text{EL} \\ \text{ENL} \end{bmatrix} \quad (61)$$

The division of variables into linear and nonlinear sets is generally arbitrary except that the division must be made to insure that the square matrix AL is not singular. Adopting the variable groupings of Bartlett and Kendall (Ref. 16) the linear and nonlinear corrections are arranged as follows:

$$\Delta\text{VL}(\Delta f_2, \Delta f_3, \dots, \Delta f_e, \Delta f'_w, \Delta f'_2, \dots, \Delta f'_e, \Delta f''_w, \Delta f''_2, \dots, \Delta f''_{e-1})$$

$$\Delta\text{VNL}(\Delta f_w, \Delta f''_e, \Delta f'''_w, \Delta f'''_2, \dots, \Delta f'''_e)$$

The linear and nonlinear equations are sequenced within the partitioned square matrix [A] as shown in Figures 1 and 2, which demonstrate the case of 4 nodes across the boundary layer.

Expanding equation (61), the resulting linear and nonlinear equations are, respectively:

$$\begin{bmatrix} \text{AL} \end{bmatrix} \begin{bmatrix} \Delta\text{VL} \end{bmatrix} + \begin{bmatrix} \text{BL} \end{bmatrix} \begin{bmatrix} \Delta\text{VNL} \end{bmatrix} = \begin{bmatrix} -\text{EL} \end{bmatrix} \quad (62)$$

$$\begin{bmatrix} \text{ANL} \end{bmatrix} \begin{bmatrix} \Delta\text{VL} \end{bmatrix} + \begin{bmatrix} \text{BNL} \end{bmatrix} \begin{bmatrix} \Delta\text{VNL} \end{bmatrix} = \begin{bmatrix} -\text{ENL} \end{bmatrix} \quad (63)$$

Solving equation (62) for the linear corrections:

AL	BL	Δf_2	Δf_3	Δf_e	Δf_w	$\Delta f'_2$	$\Delta f'_3$	$\Delta f'_e$	$\Delta f'_w$	$\Delta f''_2$	$\Delta f''_3$	$\Delta f''_e$	$\Delta f''_w$	$\Delta f'''_2$	$\Delta f'''_3$	$\Delta f'''_e$
TAYLOR	f_2	-1			$\delta\eta_{12}$					$\frac{\delta\eta_{12}^2}{2}$			$\frac{\delta\eta_{12}^3}{8}$	$\frac{\delta\eta_{12}^3}{24}$		
TAYLOR	f_3		-1			$\delta\eta_{13}$				$\frac{\delta\eta_{13}^2}{2}$				$\frac{\delta\eta_{13}^3}{24}$	$\frac{\delta\eta_{13}^3}{8}$	
TAYLOR	f_e			-1			$\delta\eta_{3e}$								$\frac{\delta\eta_{3e}^3}{8}$	$\frac{\delta\eta_{3e}^3}{24}$
B.C.	f'_w				1											
	f'_2				1	-1			$\delta\eta_{12}$				$\frac{\delta\eta_{12}^2}{3}$	$\frac{\delta\eta_{12}^2}{6}$		
	f'_3					1	-1			$\delta\eta_{13}$				$\frac{\delta\eta_{13}^2}{3}$	$\frac{\delta\eta_{13}^2}{6}$	
	f'_e						1	-1			$\delta\eta_{3e}$				$\frac{\delta\eta_{3e}^2}{3}$	$\frac{\delta\eta_{3e}^2}{6}$
	f''_2								1	-1			$\frac{\delta\eta_{12}}{2}$	$\frac{\delta\eta_{12}}{2}$		
	f''_3									1	-1			$\frac{\delta\eta_{13}}{2}$	$\frac{\delta\eta_{13}}{2}$	
	f''_e											1			$\frac{\delta\eta_{3e}}{2}$	$\frac{\delta\eta_{3e}}{2}$

FIGURE 1. LINEAR PORTION OF [A] MATRIX

ANL	BNL	Δf_2	Δf_3	Δf_e	Δf_w	$\Delta f_2'$	$\Delta f_3'$	$\Delta f_e'$	$\Delta f_w'$	$\Delta f_2''$	$\Delta f_3''$	$\Delta f_e''$	$\Delta f_w''$	$\Delta f_2'''$	$\Delta f_3'''$	$\Delta f_e'''$
B.C.	f_w								1							
MOMENTUM WALL — 2		C 1			C 4	C 3			C 2	C 5			C 8	C 7		
MOMENTUM 2 — 3		C 2	C 1			C 4	C 3			C 6	C 5			C 8	C 7	
MOMENTUM 3 — e			C 2	C 1			C 4	C 3			C 6	C 5			C 8	C 8
B.C.	f_e'							-1								
B.C.	f_e''											-1				

FIGURE 2. NONLINEAR PORTION OF [A] MATRIX

$$\begin{bmatrix} \Delta VL \end{bmatrix} = - \begin{bmatrix} AL \end{bmatrix}^{-1} \begin{bmatrix} BL \end{bmatrix} \begin{bmatrix} \Delta VNL \end{bmatrix} + \begin{bmatrix} AL \end{bmatrix}^{-1} \begin{bmatrix} -EL \end{bmatrix} \quad (64)$$

Examination of the $\begin{bmatrix} AL \end{bmatrix}$ and $\begin{bmatrix} BL \end{bmatrix}$ matrices reveals that the product $\begin{bmatrix} AL \end{bmatrix}^{-1} \begin{bmatrix} BL \end{bmatrix}$ may be performed given only the nodal spacing and must be determined only once for a given problem, thus:

$$\begin{bmatrix} \Delta VL \end{bmatrix} = - \begin{bmatrix} BAL \end{bmatrix} \begin{bmatrix} \Delta VNL \end{bmatrix} + \begin{bmatrix} ELA \end{bmatrix} \quad (65)$$

where:

$$\begin{bmatrix} BAL \end{bmatrix} = \begin{bmatrix} AL \end{bmatrix}^{-1} \begin{bmatrix} BL \end{bmatrix} \quad (66)$$

$$\begin{bmatrix} ELA \end{bmatrix} = \begin{bmatrix} AL \end{bmatrix}^{-1} \begin{bmatrix} -EL \end{bmatrix} \quad (67)$$

Applying equation (65) to equation (63), and solving for the nonlinear corrections:

$$\begin{bmatrix} \Delta VNL \end{bmatrix} = \begin{bmatrix} BNL \end{bmatrix}^{-1} \begin{bmatrix} ENL \end{bmatrix} \quad (68)$$

where:

$$\begin{bmatrix} \overline{BNL} \end{bmatrix} = \begin{bmatrix} BNL \end{bmatrix} - \begin{bmatrix} ANL \end{bmatrix} \begin{bmatrix} BAL \end{bmatrix} \quad (69)$$

$$\begin{bmatrix} \overline{ENL} \end{bmatrix} = \begin{bmatrix} -ENL \end{bmatrix} - \begin{bmatrix} ANL \end{bmatrix} \begin{bmatrix} ELA \end{bmatrix} \quad (70)$$

I. SOLUTION PROCEDURE

The matrix solution proceeds according to the following algorithm:

1. Given the nodal spacing determine $\begin{bmatrix} AL \end{bmatrix}^{-1}$ and form the product $\begin{bmatrix} BAL \end{bmatrix}$.
2. Given initial values of the primary variables determine $\begin{bmatrix} -EL \end{bmatrix}$ and $\begin{bmatrix} -ENL \end{bmatrix}$, and form the product $\begin{bmatrix} ELA \end{bmatrix}$.

3. Form $\begin{bmatrix} \overline{\text{BNL}} \\ \text{BNL} \end{bmatrix}$ and $\begin{bmatrix} \overline{\text{ENL}} \end{bmatrix}$ from the coefficients of $\begin{bmatrix} \text{ANL} \end{bmatrix}$ and $\begin{bmatrix} \text{BNL} \end{bmatrix}$.
4. Invert $\begin{bmatrix} \overline{\text{BNL}} \end{bmatrix}$ and form the product $\begin{bmatrix} \overline{\text{BNL}} \end{bmatrix}^{-1} \begin{bmatrix} \overline{\text{ENL}} \end{bmatrix}$, which is the solution for $\begin{bmatrix} \Delta \text{VNL} \end{bmatrix}$.
5. Solve for $\begin{bmatrix} \Delta \text{VL} \end{bmatrix}$ from equation (64) using $\begin{bmatrix} \Delta \text{VNL} \end{bmatrix}$ as determined in step 4.
6. Add the linear $\begin{bmatrix} \Delta \text{V} \end{bmatrix}$ to the corresponding primary variables to complete the iteration.
7. Compute the new errors $\begin{bmatrix} -\text{EL} \end{bmatrix}$ and $\begin{bmatrix} -\text{ENL} \end{bmatrix}$, and test for a maximum allowable error. If this criterion has not been reached, return to step 2 using these errors to commence the new iteration.

III. RESULTS AND DISCUSSION

The computer program has been used to investigate three basic categories of boundary layer problems using a flat plate model; steady Blasius flow (zero pressure gradient), steady Howarth flow (uniform adverse pressure gradient), and oscillating Blasius flow. The velocity profiles which resulted from these solutions are compared to previously reported results and presented in Figures 3, 4, and 5.

A. STEADY BLASIUS FLOW

In Figure 3, computed values corresponding to three nodal spacing choices are superimposed on the Blasius profile. Agreement between the numerical solution and the classical solution is better than 0.08 percent for as few as 10 nodal points.

B. STEADY HOWARTH FLOW

Results for a steady flow in a uniform adverse pressure gradient are compared with the solution of Howarth (Ref. 11) in Figure 4. The comparison of the computer solution with that of Howarth is accomplished by correlating the dimensionless parameter x^* ($=b_1 s/U_0$), which characterizes Howarth's velocity profiles. Agreement between the two solutions is generally good with differences no greater than 4.35 percent for 22 nodal points. The differences encountered may be reduced somewhat by increasing the free stream velocity of the test problem. This results in larger

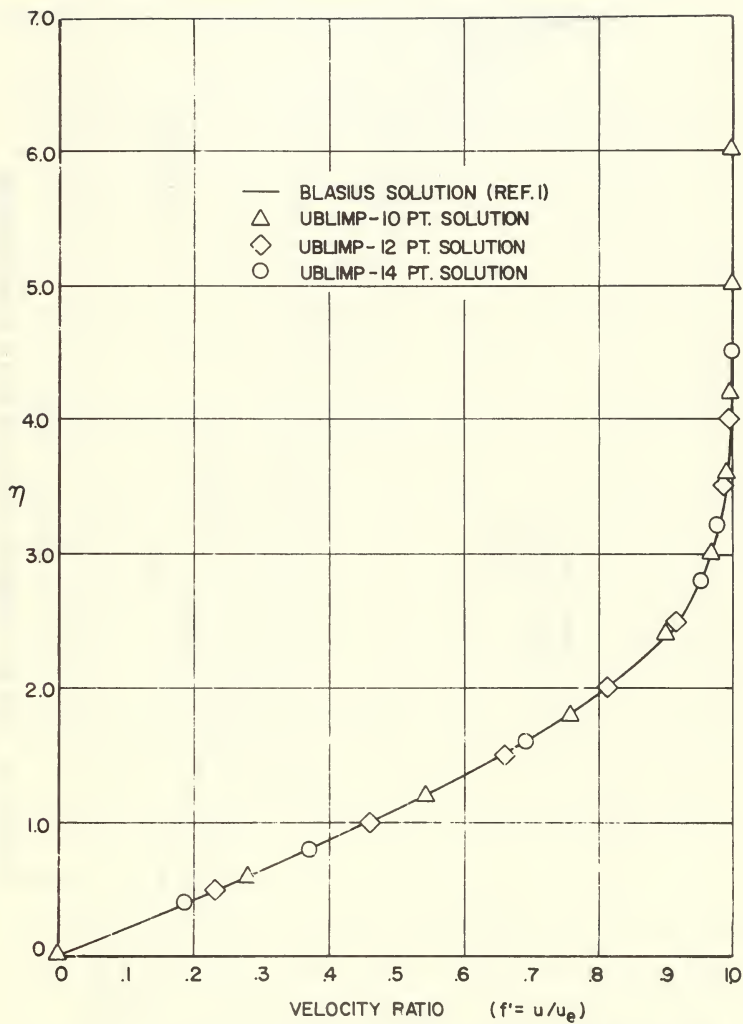


FIGURE 3. STEADY BLASIUS FLOW DATA COMPARISON

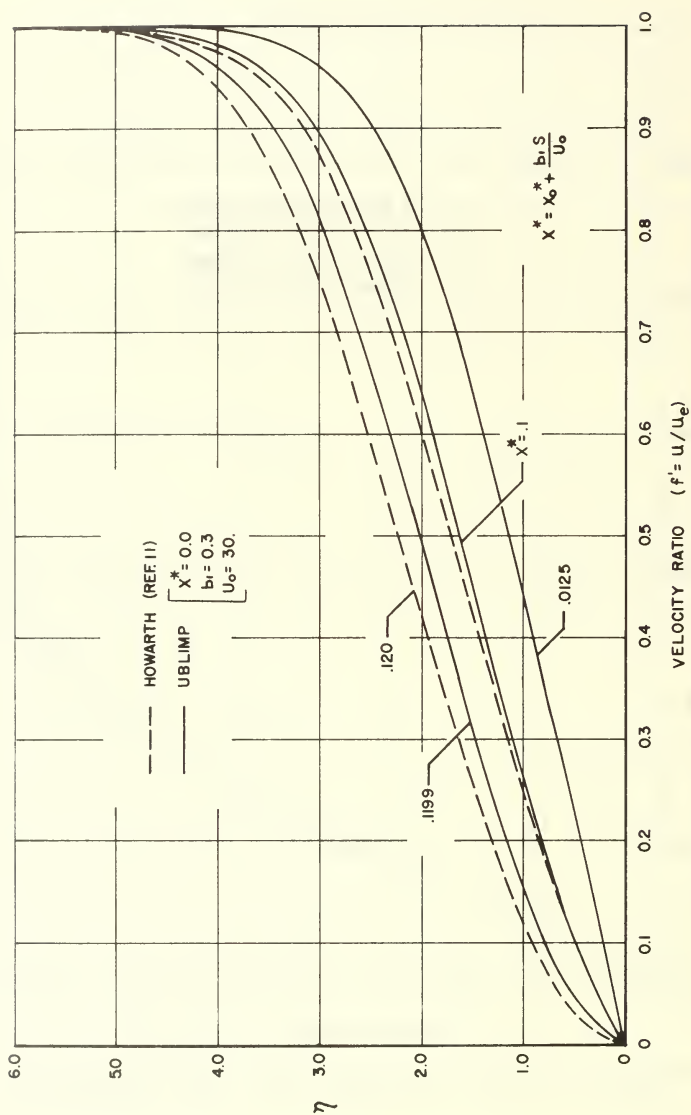


FIGURE 4. STEADY HOWARTH FLOW DATA COMPARISON

stream-wise pressure differences in the equations of motion and consequently lesser roundoff errors.

C. OSCILLATING BLASIUS FLOW

The unsteady capabilities of the numerical procedure were investigated in the solution of Blasius flow with sinusoidal oscillations superimposed on the mean flow. Utilizing the critical oscillation frequency ω_0 ($=.6U_0/s$, for a flat plate) of Lighthill's analysis (Ref. 2) to determine the solution approximation with which to compare the numerical results, the profile of velocity fluctuations which resulted was compared with the "quasi-steady" analysis as shown in Figure 5. Agreement between the calculated velocities and the Lighthill profile is within 15 percent deep in the boundary layer and within 5.5 percent in the outer regions. Noting that Lighthill's analysis is based on linearized small perturbation theory and that the amplitude of oscillation in the test problem of $0.1 U_0$ cannot be considered a small perturbation, better agreement could scarcely be expected.

D. NUMERICAL PROCEDURE

The computer program has been operated on both IBM 360/67 and CDC 6600 computer installations. Core storage requirements are 130,000 bytes for a maximum array size encompassing 25 nodal points across the boundary layer, 50 streamwise stations and 50 time increments per oscillation cycle. A

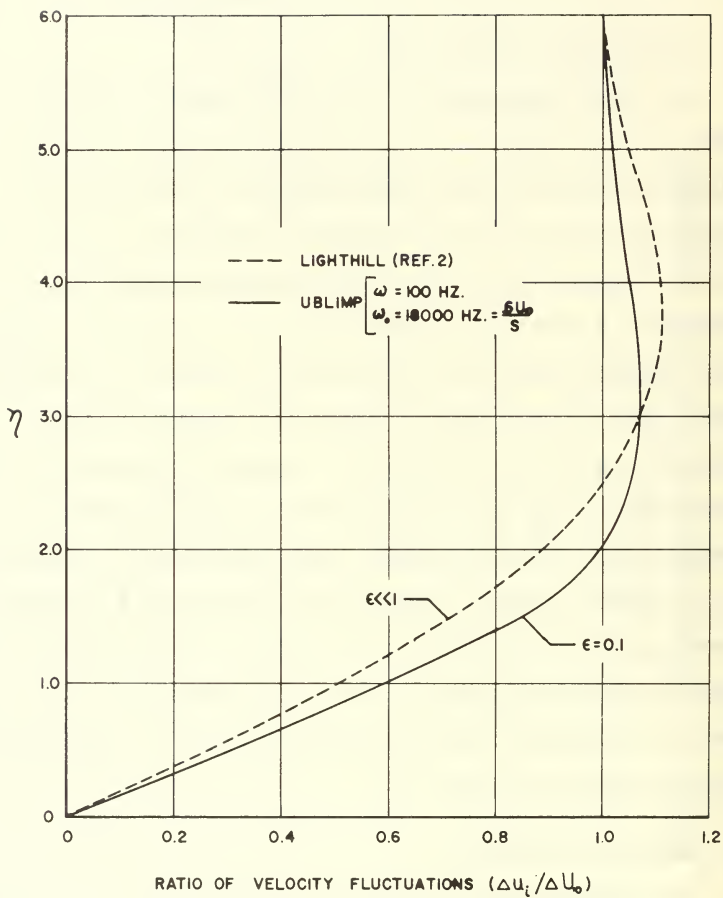


FIGURE 5. OSCILLATING BLASIIUS FLOW
DATA COMPARISON

typical 25 nodal point Blasius solution was performed in 7 seconds on the CDC 6600.

Limitations in the present numerical capabilities of the program precluded a more extensive investigation of parametric effects, however the results qualitatively demonstrated the feasibility of the computational procedure. Further refinements to the computer program should enable a more extensive investigation to include quantitative comparisons with experimental results.

IV. CONCLUSIONS AND RECOMMENDATIONS

The feasibility of the Unsteady Boundary Layer Integral Matrix Procedure in the treatment of simple steady and unsteady flow problems has been demonstrated. Favorable comparison with classical results provides a firm base for continued investigation in the unsteady regime.

Continued research utilizing the program should include the following revisions:

1. Alteration of the present time derivative component arrays (FTM1) and FTM2) to insure that all values of the components are stored for each nodal point, instead of retaining only the values at the wall.
2. Incorporation of Blasius profiles as first estimates for the primary variables in order to decrease the time required to complete the first nodal iteration procedure.
3. A study of array usage aimed at reducing the required storage and subsequent conversion to double precision computation in order to minimize round-off interference.
4. Addition of the capability of handling compressible flows in order that higher velocities may be introduced.
5. Inclusion of turbulent boundary layer capabilities perhaps through the use of eddy transport properties.

APPENDIX A

TAYLOR SERIES EXPANSIONS OF INTEGRALS (Ref. 16)

$$\int_{i-1}^i f f'' d\eta = \left[f f' \right]_{i-1}^i - \int_{i-1}^i f'^2 d\eta \quad (A1)$$

$$\int_{i-1}^i f'^2 d\eta = f_i' X M_1 + f_i'' X M_2 + f_i''' X M_3 + f_{i-1}''' X M_4 \quad (A2)$$

$$X M_1 = \delta \eta \left(f_i' - f_i'' \frac{\delta \eta}{2} + f_i''' \frac{\delta \eta^2}{8} + f_{i-1}''' \frac{\delta \eta^2}{24} \right) \quad (A3)$$

$$X M_2 = -\delta \eta^2 \left(\frac{1}{2} f_i' - f_i'' \frac{\delta \eta}{3} + f_i''' \frac{11 \delta \eta^2}{120} + f_{i-1}''' \frac{\delta \eta^2}{30} \right) \quad (A4)$$

$$X M_3 = \delta \eta^3 \left(\frac{1}{8} f_i' - f_i'' \frac{11 \delta \eta}{120} + f_i''' \frac{11 \delta \eta^2}{420} + f_{i-1}''' \frac{5 \delta \eta^2}{504} \right) \quad (A5)$$

$$X M_4 = \delta \eta^3 \left(\frac{1}{24} f_i' - f_i'' \frac{\delta \eta}{30} + f_i''' \frac{5 \delta \eta^2}{504} + f_{i-1}''' \frac{\delta \eta^2}{252} \right)$$

$$2 \int_{i-1}^i \left(f' \frac{\partial f'}{\partial \eta \partial \xi} - f'' \frac{\partial f}{\partial \eta \partial \xi} \right) d\eta = -[d_0 f f' + d_1 f_{\ell-1} f' + d_2 f_{\ell-2} f']_{i-1}^i$$

$$+ 2 d_0 [f_i' X M_1 + f_i'' X M_2 + f_i''' X M_3 + f_{i-1}''' X M_4] \\ + 2 [f_i' Z M_1 + f_i'' Z M_2 + f_i''' Z M_3 + f_{i-1}'''] \quad (A7)$$

$$Z M_1 = \delta \eta \left(Y M_1 - Y M_2 \frac{\delta \eta}{2} + Y M_3 \frac{\delta \eta^2}{8} + Y M_4 \frac{\delta \eta^2}{24} \right) \quad (A8)$$

$$Z M_2 = -\delta \eta^2 \left(\frac{1}{2} Y M_1 - Y M_2 \frac{\delta \eta}{3} + Y M_3 \frac{11 \delta \eta^2}{120} + Y M_4 \frac{\delta \eta^2}{30} \right) \quad (A9)$$

$$Z M_3 = \delta \eta^3 \left(\frac{1}{8} Y M_1 - Y M_2 \frac{11 \delta \eta}{120} + Y M_3 \frac{11 \delta \eta^2}{420} + Y M_4 \frac{5 \delta \eta^2}{504} \right) \quad (A10)$$

$$ZM_3 = \delta\eta^3 \left(\frac{1}{24} YM_1 - YM_2 \frac{\delta\eta}{3} + YM_3 \frac{5\delta\eta^2}{504} + YM_4 \frac{\delta\eta^2}{252} \right) \quad (A11)$$

$$YM_1 = d_1 f'_{\ell-1,i} + d_2 f'_{\ell-2,i} \quad (A12)$$

$$YM_2 = d_1 f''_{\ell-1,i} + d_2 f''_{\ell-2,i} \quad (13)$$

$$YM_3 = d_1 f'''_{\ell-1,i} + d_2 f'''_{\ell-2,i}$$

$$YM_4 = d_1 f'''_{\ell-1,i-1} + d_2 f'''_{\ell-2,i-1} \quad (A15)$$

APPENDIX B

BOUNDARY LAYER EQUATION CORRECTION COEFFICIENTS

$$C1 = (1+d_o) f'_i - \frac{\rho u}{\alpha^* 2} \{T_o (1-\ell n u_e) - (T_1 \ell n u_{e_{k-1}} + T_2 \ell n u_{e_{k-2}})\} \quad (B1)$$

$$C2 = -(1+d_o) f'_{i-1} + \frac{\rho u}{\alpha^* 2} \{T_o (1-\ell n u_e) - (T_1 \ell n u_{e_{k-1}} + T_2 \ell n u_{e_{k-2}})\} \quad (B2)$$

$$C3 = (1+d_o) f_i + (d_1 f_{\ell-1,i} + d_2 f_{\ell-2,i}) - 2(1+\beta+2d_o) X M_1 - 2Z M_1 \quad (B3)$$

$$C4 = -(1+d_o) f_{i-1} - (d_1 f_{\ell-1,i-1} + d_2 f_{\ell-2,i-1}) \quad (B4)$$

$$C5 = 1 - 2(1+\beta+2d_o) X M_2 - 2Z M_2 \quad (B5)$$

$$C6 = -1 \quad (B6)$$

$$C7 = -2(1+\beta+2d_o) X M_3 - 2Z M_3 \quad (B7)$$

$$C8 = -2(1+\beta+2d_o) X M_4 - 2Z M_4 \quad (B8)$$

APPENDIX C
PROGRAM DESCRIPTION

A. GENERAL

The computer program referred to as UBLIMP, is written in FORTRAN IV source language and has been operated on both IBM 360/67 and CDC 6600 computer installations. Core storage requirements are 130,000 bytes for a maximum array size encompassing 25 nodal points across the boundary layer, 50 streamwise stations and 50 time increments per oscillation cycle.

The program consists of a main program (UBLIMP) and 21 subroutines which are divided into two sections as shown in Figure 6. LINK 0, common to both sections, contains the main program and certain service routines which are common to both sections in the fully expanded program. LINK 1 sets up the boundary conditions and LINK 2 iterates to solve the boundary layer equations.

The program may be applied to steady or unsteady flow, two-dimensional cartesian or axisymmetric bodies with similar or nonsimilar profiles. It allows quadratic or cubic curve fits of the primary variables. Program options are controlled by the control variable KR and are detailed in the input instructions for the UBLIMP program in Appendix D. A program linkage schematic is shown in Figure 7 and a functional flow chart is shown in Figure 8. Correlation

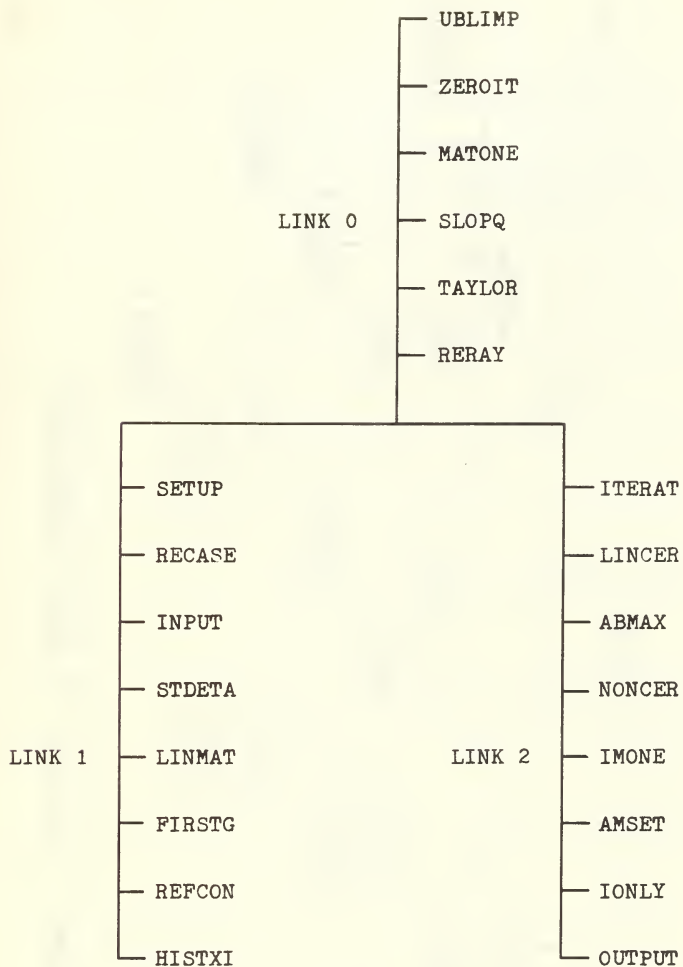


FIGURE 6. COMPUTER PROGRAM OVERLAY STRUCTURE

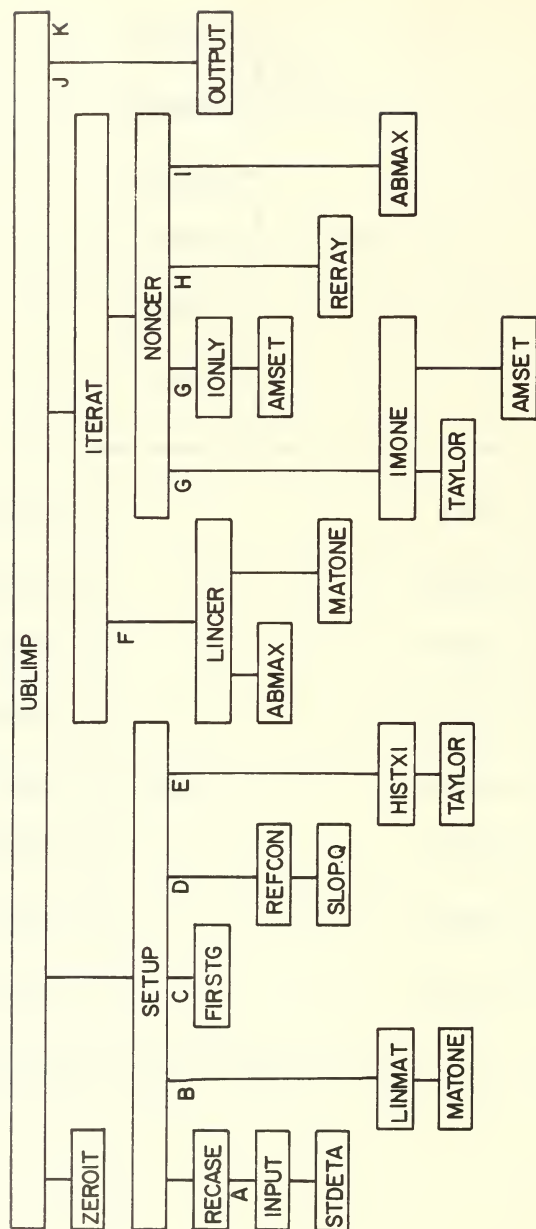


FIGURE 7. SCHEMATIC OF UBLIMP SUBROUTINES

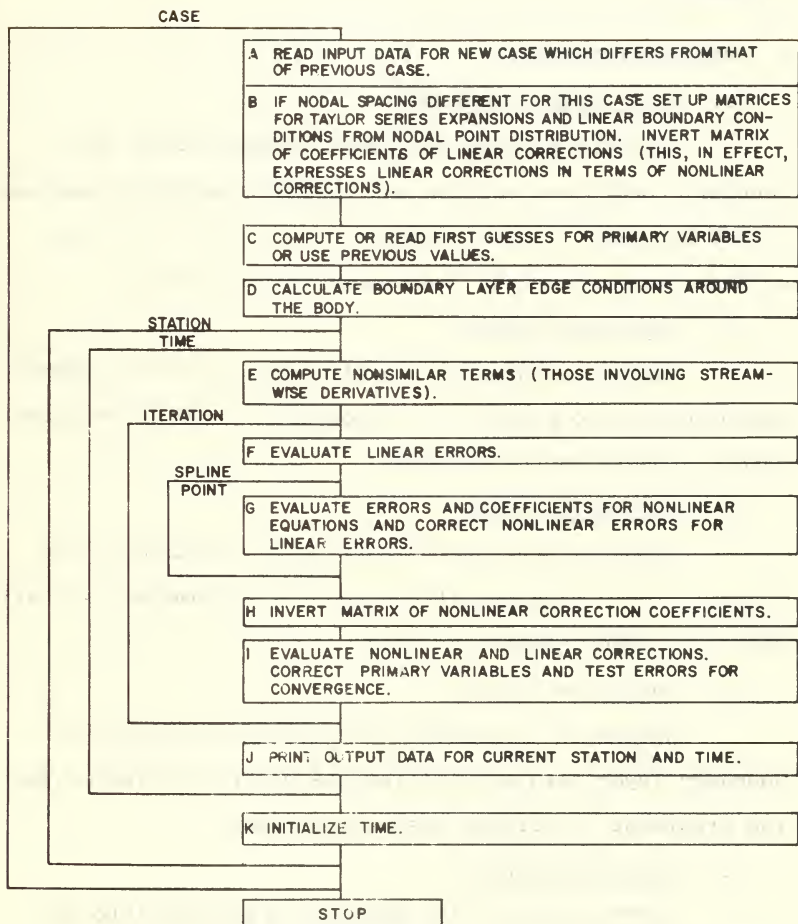


FIGURE 8. FUNCTIONAL FLOW CHART OF UBLIMP PROGRAM

between the two is indicated by letters running from A through K.

B. PROGRAM FUNCTIONS

1. Main Program (UBLIMP)

The main program provides linkage between the boundary conditions section and iterative solution section of the program. Initial core zeroing and the call for solution output are also initiated by the routine.

2. Subroutine ZEROIT

ZEROIT performs the initial zeroing of all named common blocks to eliminate the possibility of introducing random numbers in the solution.

3. Subroutine SETUP

SETUP is the control routine for setting up the boundary layer edge conditions and the streamwise derivatives for a new time, station or case.

4. Subroutine RECASE

RECASE is the control routine for the input of boundary layer data and initializes control variables for the treatment of surface discontinuities.

5. Subroutine INPUT

INPUT reads all the data for a problem into the computer.

6. Subroutine STDETA

STDETA permits simplification of the input data deck by providing standard nodal point distributions should the user not care to provide one.

7. Subroutine LINMAT

LINMAT sets up the matrices for the Taylor series expansions and linear boundary conditions from the eta spacing and solves to express the linear corrections in terms of the nonlinear corrections.

8. Subroutine MATONE

MATONE performs operations on a column of a matrix of coefficients $[BL]$ or on a column of a matrix of errors $[-EL]$ (designated E in the subroutine call list) such as to form $[AL]^{-1}[E]$.

9. Subroutine FIRSTG

FIRSTG computes first estimates of the primary variables based on the nodal point distribution should the user not care to provide them.

10. Subroutine REFCON

REFCON calculates the boundary layer edge conditions and sets up the wall boundary conditions.

11. Subroutine SLOPQ

SLOPQ defines a cubic equation for a set of points, calculates the average slope at each point and integrates the cubic equation between each pair of points.

12. Subroutine HISTXI

HISTXI computes terms involving the axial derivatives and time derivatives and stores those upstream quantities needed for these difference relations.

13. Subroutine TAYLOR

TAYLOR calculates the coefficients in the Taylor series expansions of the integrals which appear in the integral form of the boundary layer equations.

14. Subroutine ITERAT

ITERAT is the control program for performing the boundary layer iteration.

15. Subroutine LINCER

LINCER evaluates errors for the linear equations and determines the maximum linear error.

16. Subroutine ABMAX

ABMAX searches a given array for the entry with the maximum absolute value.

17. Subroutine NONCER

NONCER performs that portion of a boundary layer iteration having to do with the solution of the nonlinear corrections of the nonlinear equations, computes the damping factor and applies it to the corrections and then applies the damped corrections to the primary variables.

18. Subroutine IMONE

IMONE evaluates the coefficients of $(i-1)$ corrections for the i^{th} nonlinear equations, where i is the i^{th} nodal point in the boundary layer.

19. Subroutine AMSET

AMSET calculates the contributions of the $\begin{bmatrix} \text{ANL} \end{bmatrix}$ and applies them to $\begin{bmatrix} \text{BNL} \end{bmatrix}$ and $\begin{bmatrix} \text{ENL} \end{bmatrix}$.

20. Subroutine IONLY

IONLY evaluates the coefficients of the i^{th} corrections for the i^{th} nonlinear equations, where i is the i^{th} nodal point in the boundary layer.

21. Subroutine RERAY

RERAY replaces a matrix $[C]$ with its inverse and forms the product of the inverse with another matrix $[D]$.

22. Subroutine OUTPUT

OUTPUT prints a standard boundary layer output block for a converged solution, or at the end of each iteration if desired.

APPENDIX D

INPUT INSTRUCTIONS FOR THE UBLIMP PROGRAM

GROUP 1 CONTROL CARD, TITLE

CARD 1 FORMAT(20I1,15A4)

FIELD 1 (Columns 1-20) This is the variable KR which is used to control the various program options.

COLUMN 1 Determines whether a new set of ETA values is to be input for the present case.

0 Uses resident values from previous case.

1 Values input by user (mandatory for first case).

COLUMN 2 Designates the type of first guesses to be utilized for primary variables.

0 Uses built-in relations to calculate first guesses.

1 First guesses input by user.

2 Uses resident values from previous case (cannot be used for first case).

COLUMN 3 Determines the treatment of streamwise derivatives.

0 Performs similar solution at each streamwise station.

1 Considers two-point difference relations at all stations except that a similar solution is performed at the first station for non-blunt bodies and at the first two stations for blunt bodies.

2 Considers three-point difference relations at all stations except that a similar solution is performed at the first station and a two-point solution is performed at the second station for non-blunt bodies; similar solutions are performed at the first and second stations and a two-point solution is performed at the third station for blunt bodies; and a two-point solution is performed for the first station after a discontinuity.

COLUMN 4 Determines when output block is to be printed.

- 0 Output block printed for converged solution or for nonconverged solution after 50 iterations.
- 1 Output block printed after each iteration.

COLUMN 5 Determines treatment of Entropy Layer.
(Not used in present program. KR(5)=4 is input to maintain input data sequencing.)

COLUMN 6 Designates body shape.

- 0 Axisymmetric blunt body.
- 1 Planar blunt body
- 2 Axisymmetric sharp body.
- 3 Planar sharp body.
- 4 Axisymmetric or planar shape which has no sharp tip or blunt stagnation point, such as a nozzle.

COLUMN 7 Steady, zero pressure-gradient noise filter.

- 0 Normal computations of DUDS.
- 1 DUDS is forced to zero.

COLUMN 8 Designates form in which wall mass fluxes are input. KR(8) is not utilized if wall mass fluxes are not input.

- 0 Wall fluxes input in LBS/SEC FT**2
- 1 Wall fluxes input in normalized form (divided by -ALPHASTAR).

COLUMN 9 Together with KR(11) this designates the type of wall boundary conditions.

- 0 Assigned stream function at the wall.
- 1 Assigned mass flux at the wall. KR(9)=1 is used if zero mass flux is used.

COLUMN 10 Determines the type of curve fits employed to represent the primary variables.
(KR(10)=0 is recommended for accuracy for most problems, however for severe problems KR(10)=1 is better since the cubics can become poorly behaved.)

- 0 Utilizes connected cubics.
 - 1 Utilizes connected quadratics except for the outermost segment where connected cubics are used.
- COLUMN 11 Together with KR(9) this designates the type of wall boundary conditions. (Presently KR(11)=0 is the only active value and requires an assignment of wall temperature.)
- COLUMN 12 Provides access to standard nodal-point distributions built into the program.
- 0 User inputs data for nodal-points.
 - 1 8 points with more concentration at the wall.
 - 2 8 points equally spaced.
 - 3 10 points equally spaced.
 - 4 12 points equally spaced.
 - 5 14 points equally spaced.
 - 6 15 points with more concentration at the wall.
 - 7 18 points equally spaced.
 - 8 22 points equally spaced.
 - 9 25 points equally spaced.
- COLUMN 13 Permits the assignment of a convergence damping factor. (This factor is overridden if a smaller damping factor is computed internally by some constraint.)
- 0 No damping factor is assigned.
 - J If J is greater than zero, corrections are damped uniformly by J/10.
- COLUMN 14 Non-zero entry causes a complete set of primary variables to be output for future use as first guess inputs.
- COLUMN 15 Non-zero entry provides debug output for first guesses and linear matrices.

0 No debug.

1 First guesses are dumped.

2 Linear matrices before and after inversion are also dumped.

COLUMN 16 Non-zero input indicates the pressure profile has been input in the form of pressure coefficients.

COLUMN 17 Non-zero entry provides debug output for coefficients.

0 No debug.

Y For (Y+1-ITS) greater than zero, where ITS is the number of the current boundary layer iteration, the coefficients which combine to make up the nonlinear equations are dumped and the derivatives of the nonlinear equations with respect to the nonlinear variables are dumped before and after inversion.

COLUMN 18 Not used.

COLUMN 19 Non-zero entry provides debug output for linear and nonlinear errors.

COLUMN 20 Determines which of the two internal sets of first guesses are to be used if the user does not provide them.

0 Straight line Blasius inputs are calculated for the stream function and velocity ratio, and zero is input for the first and second derivatives of the velocity ratio.

1 Straight line Blasius inputs are calculated for all the primary variables.

FIELD 2 (Columns 21-80) CASE. Title of the case (ALPHA-NUMERIC). Used for identification of printed output.

GROUP 2 TIMES AND STATIONS

CARD 1 FORMAT(I3)

FIELD 1 (Columns 1-3, right-justified) NITEM Number of time grid points per cycle when considering an unsteady solution, otherwise input 1. (max=50)

CARD 2 FORMAT(E10.4)

FIELD 1 (Columns 1-10) TIME(1) Value to be used in
the output block to identify a time of solution.

CARD 3 FORMAT(I3)

FIELD 1 (Columns 1-3, right-justified) NS. Number of
streamwise stations. (max=50)

CARD 4 FORMAT(8E10.4)

FIELD 1,2,...8 per card(10 columns per field) S(IS)
Streamwise distance upon which the boundary
layer solution is based in feet. Blunt body
problems should start with S=0.0. Sharp body
or nozzle problems must start with some finite
value of S. The boundary layer is assumed
similar up to and including this first station.
A negative entry for S indicates a discon-
tinuity at that station, and alters the differ-
encing scheme for axial derivatives.

GROUP 3 NODAL DATA (Skip this GROUP for KR(12)≠ 0)

CARD 1 FORMAT(I3)

FIELD 1 (Columns 1-3, right-justified) NETA Number of
nodal points across the boundary layer.
(maximum of 25)

CARD 2 FORMAT(8E10.4)

FIELD 1,2,...8 per card(10 columns per field) ETA(I)
ETA stations across the boundary layer starting
at the wall (ETA=0.0). It is recommended that
a value between 5.0 and 6.0 be assigned at the
boundary layer edge. UBLIMP values of ETA are
equivalent to Blasius values divided by the
square root of two.

GROUP 4 BODY SHAPE DATA

CARD 1 FORMAT(2E10.4) USED ONLY IF KR(6)=0 or 1.

FIELD 1 (Columns 1-10) CONE. Cone half angle in sphere-
cone shape bodies. Leave blank for other body
shapes.

FIELD 2 (Columns 11-20) RNOSE. Effective nose radius
in feet. Used to calculate stagnation point
velocity gradient from Newtonian relations.

CARD 2 FORMAT(8E10.4) USED ONLY IF KR(6)=0, 2 or 4

FIELD 1,2,...8 per card(10 columns per field) ROKAP(IS)
This is the local body radius in feet normal to the body centerline raised to the n power, where n is unity for axisymmetric bodies and zero for planar bodies. Therefore, ROKAP is unity for planar bodies and the local body radius for axisymmetric bodies. For planar bodies, this card set is used only if KR(6)=4. (A special input format can be used for spherical-nosed bodies as follows. Set ROKAP(1) equal to minus the nose radius. The nose radius is then set to -ROKAP(1) and ROKAP(1) is set to zero. If subsequent ROKAP are input as zeros, the program computes ROKAP from S for a spherical nose. The first non-zero entry is then ROKAP at the tangent point. If this again followed by zeros, linear interpolation is used to the next non-zero entry to yield ROKAP along a conical afterbody.)

GROUP 5 STAGNATION DATA

CARD 1 FORMAT(E10.4)

FIELD 1 (Columns 1-10) PTET. Local stagnation pressure in atmospheres.

GROUP 6 FIRST GUESS INFORMATION

CARD 1 FORMAT(5(2XE14.7))

FIELD 1,2,...5 per card(16 columns per field) F(I,J).
First guesses for stream function (F(1,J)), velocity ratio (F(2,J)), shear function (F(3,J)), and derivative of shear function (F(4,J)) across the boundary layer.

GROUP 7 STREAMWISE DISTRIBUTIONS FOR EDGE CONDITIONS

CARD 1 FORMAT(5(2XE14.7))

FIELD 1,2,...5 per card(16 columns per field) PRE(IS).
Ratio of local static to stagnation pressure for the mean flow condition. When KR(16) is non-zero PRE is input in the form of negative pressure coefficients referred to PTET.

CARD 2 FORMAT(Blank) This card is inserted to maintain data continuity. In more complex formulations this card corresponds to a control flag for updating the pressure profile during the computation.

GROUP 8 STREAMWISE DISTRIBUTIONS FOR WALL INPUT CONDITIONS

CARD 1 FORMAT(E10.4)

FIELD 1 (columns 1-10) T. Wall temperature in degrees R (for the present this is the free stream static temperature).

CARD 2 FORMAT(8E10.4) USED ONLY FOR KR(9)=0

FIELD 1,2,...8 per card(10 columns per field) FW(IS).
Wall stream function (negative for mass addition).

CARD 3 FORMAT(Blank) USED ONLY IF CARD 2 IS USED. This card is inserted to maintain data continuity. In more complex formulations this card corresponds to a control flag for updating the wall stream function profile during the computation.

CARD 4 FORMAT(8E10.4) USED ONLY FOR KR(9)=1

FIELD 1,2,...8 per card(10 columns per field) RHOVW(IS).
Total mass flux at the wall (LB/SEC FT**2 or dimensionless for KR(8)=0 or 1 respectively).
These values are positive for mass injection.

CARD 5 FORMAT(Blank) USED ONLY IF CARD 4 IS USED. This card is inserted to maintain data continuity. In more complex formulations this card corresponds to a control flag for updating the total mass flux profile during the computation.

GROUP 9 FREE STREAM DATA

CARD 1 FORMAT(3E10.4)

FIELD 1 (Columns 1-10) UMFS. Mean free stream velocity in FT/SEC.

FIELD 2 (Columns 11-20) FREQ. Frequency of mean free stream oscillations in HZ.

FIELD 3 (Columns 21-30) EPS. Ratio of the amplitude of the free stream oscillation to the local mean free stream velocity.

LAST CARD FORMAT(A1) The purpose of this entry is to permit a test on whether or not a new case is to follow. In the event a case does not converge in the allotted number of iterations, any remaining cards for that case are read and then

ignored until either a period or a comma is encountered in column 1. A comma in column 1 indicates that another case is to follow, while a period in column 1 indicates that this is the last card in the input deck.

The preceding input instructions should permit the user to fully exercise the UBLIMP program with a minimum of difficulty. For more detailed information on various routines the reader is referred to reference 18, which is the user manual prepared for the BLIMP program.

SAMPLE COMPUTER PROBLEM

The UBLIMP computer program was exercised for steady Blasius flow, steady Howarth flow and oscillating Blasius flow. The solution of steady Blasius flow for 8 nodal points across the boundary layer and 6 streamwise stations is presented as a sample problem. Solution time was 3.93 seconds utilizing the time sharing capabilities of the Naval Postgraduate School IBM 360/67 computer system.

INPUT DATA DECK

```

10204300100200000001
1
6
1.0 .001 .2 .4 .6 .8 1.0
.9994944 .9994944 .9994944 .9994944
.9994944
(blank)
520.
(blank)
(blank)
30.
.
STEADY BLASIIUS FLOW.
.9994944

```

PRINTED OUTPUT

UNSTEADY BOUNDARY LAYER INTEGRAL MATRIX PROCEDURE

DEPARTMENT OF AERONAUTICS
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

INVESTIGATION OF

STEADY BLASIO FLOW.

CONTROL NUMBERS

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	2	0	4	3	0	0	1	0	0	2	0	0	0	0	0	0	0	1

ETA VALUES

0.0	8.000E-01	1.600E 00	2.400E 00	3.200E 00	4.000E 00	5.000E 00
6.000E 00						

TEMPERATURE

(DEG R)
5.200E 02

TOTAL
PRESSURE
(ATM)
1.000E 00

MEAN
VELOCITY
(FT/SEC)
3.000E 01

OSCIL.
AMPLITUDE
PARAMETER
0.0

OSCIL.
FREQUENCY
(HERTZ)
0.0

DENSITY

VISCOSITY
TIME GRID
CYCLE TIME

=0.07650995 LB/FT CU
= 3.719E-07 LB SEC/FT SQ
= 0.0 SEC
= 0.0 SEC

STATION INPUT DATA

DISTANCE, FT	1.000E-03	2.000E-01	4.000E-01	6.000E-01	8.000E-01
	1.000E 00				
ROKAP	1.000E 00	1.000E 00	1.000E 00	1.000E 00	1.000E 00
	1.000E 00				
PRESSURE RATIO, MEAN	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01
	9.995E-01				
ENTROPY DROP, BTU/LB R	0.0	0.0	0.0	0.0	0.0
	0.0				
MASS FLUX, LB/SEC FT**2	0.0	0.0	0.0	0.0	0.0
	0.0				

STATION COMPUTED DATA TIME= 0.0

STREAMWISE DIMENSION (FEET)	XI (LB/ SEC)**2	BETA	EDGE VELOCITY (FT/SEC)	STATIC PRESSURE (ATM)
1.000E-03	8.536E-10	-3.052E-09	3.000E 01	9.995E-01
2.000E-01	1.707E-07	2.035E-07	3.000E 01	9.995E-01
4.000E-01	3.414E-07	-1.017E-07	3.000E 01	9.995E-01
6.000E-01	5.122E-07	-2.747E-06	3.000E 01	9.995E-01
8.000E-01	6.829E-07	4.069E-07	3.000E 01	9.995E-01
1.000E 00	8.536E-07	1.628E-05	3.000E 01	9.995E-01

TIME 0.0 SECONDS * * * STREAMWISE DIMENSION 0.10000E-02 FEET * * * *

ITS	ITERATED VALUES		DAMP	MAX LINEAR		MAX MOMENTUM	
	TIME	FPPW		ERROR	ERROR	ERROR	ERROR
1	0.0	0.6177	0.0852	7	1.000E-00	4	-3.976E-01
2	0.0	0.6970	0.1597	7	9.148E-01	4	-3.300E-01
3	0.0	0.7187	0.2736	7	7.687E-01	4	-2.231E-01
4	0.0	0.6564	0.4665	7	5.584E-01	5	-1.025E-01
5	0.0	0.5008	0.9494	7	2.979E-01	3	9.781E-02
6	0.0	0.4699	1.0000	7	1.507E-02	4	4.533E-02
7	0.0	0.4695	1.0000	4	9.537E-07	5	1.790E-03
8	0.0	0.4695	1.0000	6	1.907E-06	8	-3.551E-06

EDGE		STATIC		XI		FLUX NORM		ROKAP	
VELOCITY	(FT/SEC)	PRESSURE	(LB/	SEC)*2	BETA	PARAMETER	(FT)		
3.000E 01		9.995E-01	8.536E-10	0.0		2.066E-02	1.000E 00		

WALL FR.	COEFF,	CF	MOM TRANS	COEFF,	RHO*UE*CF/2	WALL	MASS FLUXES		TOTAL
							MECH	REM	GAS
8.451E-03			9.698E-03			(LB/SQ FT)	(LB/SQ FT)	SEC	
						9.043E-03	-0.0	0.0	

REYNOLDS	NUMBER	STROUHAL	NUMBER	MOMENTUM	THICKNESS,	THETA	(FT)	DISPLAC.	THICKNESS,	DELSTAR/	SHAPE
6.172E 03		0.0		8.451E-06				2.190E-05			2.592E 00

NODAL INFORMATION

DISTANCE FROM WALL (FT)	ETA	STREAM FUNCTION (F)	VELOCITY RATIO (FP=U/UE)	SHEAR FUNCTION (FPP)	FPPP	VELOCITY (FT/SEC)
0.0	0.0	0.0	0.0	4.695E-01	6.403E-03	0.0
1.440E-05	8.000E-01	1.495E-01	3.713E-01	4.511E-01	-5.229E-02	1.114E 01
2.880E-05	1.600E 00	5.829E-01	6.976E-01	3.420E-01	-2.205E-01	2.093E 01
4.320E-05	2.400E 00	1.232E 00	9.011E-01	1.676E-01	-2.154E-01	2.703E 01
5.761E-05	3.200E 00	1.991E 00	9.801E-01	4.695E-02	-8.628E-02	2.940E 01
7.201E-05	4.000E 00	2.784E 00	9.977E-01	6.894E-03	-1.386E-02	2.993E 01
9.001E-05	5.000E 00	3.783E 00	1.000E 00	-1.719E-05	3.861E-05	3.000E 01
1.080E-04	6.000E 00	4.783E 00	1.000E 00	1.029E-09	-4.238E-06	3.000E 01

TIME 0.0 SECONDS * * * STREAMWISE DIMENSION 0.2000E 00 FEET * * * *

ITERATED VALUES		DAMP	MAX LINEAR ERROR		MAX MOMENTUM ERROR	
ITS	TIME	FPFW				
1	0.0	0.4695	1.0000	3	1.907E-06	4 -4.897R-06
EDGE						
VELOCITY (FT/SEC)	STATIC PRESSURE (ATM)	XI (LB/SEC)**2	BETA	FLUX NORM PARAMETER	ROKAP	(FT)
3.000E 01	9.995E-01	1.707E-07	0.0	1.461E-03	1.000E 00	
WALL FR. COEFF, CF						
	MOM TRANS COEFF, RHO*UE*CF/2	WALL SHEAR (LB/SQ FT)	MASS FLUXES MECH REM (LB/SQ FT SEC)		TOTAL GAS	
5.976E-04	6.858E-04	6.394E-04	-0.0		0.0	
REYNOLDS NUMBER						
	STROUHAL NUMBER	MOMENTUM THICKNESS, THETA (FT)	DISPLACE. THICKNESS, DELSTAR (FT)	SHAPE FACTOR, DELSTAR/THETA		
1.234E 06	0.0	1.195E-04	3.098E-04	2.592E 00		

NODAL INFORMATION

DISTANCE FROM WALL (FT)	ETA	STREAM FUNCTION (F)	VELOCITY RATIO (FP=U/UE)	SHEAR FUNCTION (FPP)	FPPP	VELOCITY (FT/SEC)
0.0	0.0	0.0	0.0	4.695E-01	6.398E-03	0.0
2.037E-04	8.000E-01	1.495E-01	3.714E-01	4.511E-01	-5.229E-02	1.114E 01
4.073E-04	1.600E 00	5.829E-01	6.976E-01	3.420E-01	-2.205E-01	2.093E 01
6.110E-04	2.400E 00	1.232E 00	9.011E-01	1.676E-01	-2.154E-01	2.703E 01
8.147E-04	3.200E 00	1.991E 00	9.801E-01	4.695E-02	-8.627E-02	2.940E 01
1.018E-03	4.000E 00	2.784E 00	9.977E-01	6.894E-03	-1.386E-02	2.993E 01
1.273E-03	5.000E 00	3.783E 00	1.000E 00	-1.782E-05	3.560E-05	3.000E 01
1.528E-03	6.000E 00	4.783E 00	1.000E 00	2.059E-09	5.272E-08	3.000E 01

TIME 0.0 SECONDS * * * STREAMWISE DIMENSION 0.4000E 00 FEET * * * *

ITERATED VALUES	DAMP	MAX LINEAR	MAX MOMENTUM
ITS	FPW	ERROR	ERROR
1	0.4695	3 9.537E-07	8 9.759E-06

EDGE	STATIC	XI	BETA	FLUX NORM	ROKAP
VELOCITY	PRESSURE	(LB/	SEC)**2	PARAMETER	(FT)
(FT/SEC)	(ATM)				
3.000E 01	9.995E-01	3.414E-07	0.0	1.033E-03	1.000E 00

WALL FR.	MOM TRANS	WALL	MASS FLUXES
COEFF,	COEFF,	SHEAR	MECH
CF	RHO*UE*CF/2		REM
		(LB/SQ FT)	GAS
4.226E-04	4.849E-04	4.521E-04	(LB/SQ FT SEC)
		-0.0	0.0

REYNOLDS	STROUHAL	MOMENTUM	DISPLACE.	SHAPE
NUMBER	NUMBER	THICKNESS,	THICKNESS,	FACTOR,
		THETA	DELSTAR	DELSTAR/
		(FT)	(FT)	THETA
2.469E 06	0.0	1.690E-04	4.381E-04	2.592E 00

NODAL INFORMATION

DISTANCE	STREAM	VELOCITY	SHEAR	FPPP	VELOCITY
FROM WALL	FUNCTION	RATIO	FUNCTION		(FT/SEC)
(FT)	(F)	(FP=U/UE)	(FPP)		
0.0	0.0	0.0	4.695E-01	6.420E-03	0.0
2.880E-04	8.000E-01	3.714E-01	4.511E-01	-5.230E-02	1.114E 01
5.761E-04	1.600E 00	6.976E-01	3.420E-01	-2.205E-01	2.093E 01
8.641E-04	2.400E 00	9.011E-01	1.676E-01	-2.154E-01	2.703E 01
1.152E-03	3.200E 00	9.801E-01	4.695E-02	-8.627E-02	2.940E 01
1.440E-03	4.000E 00	9.977E-01	6.894E-03	-1.387E-02	2.993E 01
1.800E-03	5.000E 00	1.000E 00	-1.739E-05	4.356E-05	3.000E 01
2.160E-03	6.000E 00	1.000E 00	4.119E-09	-8.768E-06	3.000E 01

TIME 0.0 SECONDS * * * STREAMWISE DIMENSION 0.6000E 00 FEET * * * *

ITERATED VALUES		DAMP	MAX LINEAR	MAX MOMENTUM		
ITS	TIME	FPPW	ERROR	ERROR		
1	0.0	0.4695	3	9.537E-07	8	-2.644E-05

EDGE	STATIC	XI	FLUX NORM	ROKAP	
VELOCITY	PRESSURE	(LR/	PARAMETER	(FT)	
(FT/SEC)	(ATM)	SEC)**2			
3.000E 01	9.995E-01	5.122E-07	0.0	8.434E-04	1.000E 00

WALL FR.	MOM TRANS	WALL	MASS FLUXES
COEFF,	COEFF,	SHEAR	MECH
CF	RHO*UE*CF/2	(LB/SQ FT)	REM
3.450E-04	3.959E-04	3.692E-04	(LB/SQ FT SEC)
		-0.0	0.0

REYNOLDS	STROUHAL	MOMENTUM	DISPLACE.	SHAPE
NUMBER	NUMBER	THICKNESS,	THICKNESS,	FACTOR,
		THETA	DELSTAR,	DELSTAR/
		(FT)	(FT)	THETA
3.703E 06	0.0	2.070E-04	5.365E-04	2.592E 00

NODAL INFORMATION

DISTANCE	STREAM	VELOCITY	SHEAR	VELOCITY
FROM WALL	FUNCTION	RATIO	FUNCTION	(FT/SEC)
(FT)	(F)	(FP=U/UE)	(FPP)	
0.0	0.0	0.0	4.695E-01	6.350E-03
3.528E-04	1.495E-01	3.713E-01	4.511E-01	-5.224E-02
7.055E-04	1.600E 00	6.976E-01	3.420E-01	-2.206E-01
1.058E-03	2.400E 00	9.011E-01	1.676E-01	-2.153E-01
1.411E-03	3.200E 00	9.801E-01	4.695E-02	-8.632E-02
1.764E-03	4.000E 00	9.977E-01	6.891E-03	-1.383E-02
2.205E-03	5.000E 00	1.000E 00	-1.543E-05	1.958E-05
2.646E-03	6.000E 00	1.000E 00	8.235E-09	1.131E-05
				3.000E 01

TIME 0.0 SECONDS * * * STREAMWISE DIMENSION 0.80000E 00 FEET * * * *

ITERATED VALUES		DAMP		MAX LINEAR		MAX MOMENTUM	
ITS	TIME	FPPW		ERROR		ERROR	
1	0.0	0.4695	1.000	4	9.537E-07	8	-6.526E-05
EDGE		STATIC		XI		FLUX NORM	
VELOCITY	PRESSURE	(LB/	SEC)**2	BETA	PARAMETER	ROKAP	(FT)
(FT/SEC)	(ATM)						
3.000E 01	9.995E-01	6.829E-07	0.0	7.304E-04	1.000E 00		
WALL FR.		MOM TRANS		WALL		MASS FLUXES	
COEFF,	CF	COEFF,	RHO*UE*CF/2	SHEAR		MECH	TOTAL
				(LB/SQ FT)		REM	GAS
2.988E-04	3.429E-04			3.197E-04	-0.0	(LB/SQ FT SEC)	0.0
REYNOLDS		STROUHAL		MOMENTUM		SHAPE	
NUMBER		NUMBER		THICKNESS,	DISPLACE.	THICKNESS,	FACTOR,
				THETA	DELSTAR,	DELSTAR,	DELSTAR,
				(FT)	(FT)	THETA	THETA
4.937E 06	0.0	2.390E-04	6.195E-04	2.592E 00			

NODAL INFORMATION

DISTANCE		STREAM		VELOCITY		SHEAR		VELOCITY	
FROM WALL	ETA	FUNCTION		RATIO		FUNCTION	FPPP	(FT/SEC)	
(FT)		(F)		(FP=U/UE)		(FPP)			
0.0	0.0	0.0		0.0		4.695E-01	6.411E-03	0.0	
4.073E-04	8.000E-01	1.495E-01	3.714E-01	4.511E-01	-5.229E-02	1.114E 01			
8.147E-04	1.600E 00	5.829E-01	6.976E-01	3.420E-01	-2.205E-01	2.093E 01			
1.222E-03	2.400E 00	1.232E 00	9.011E-01	1.676E-01	-2.154E-01	2.703E 01			
1.629E-03	3.200E 00	1.991E 00	9.801E-01	4.694E-02	-8.632E-02	2.940E 01			
2.037E-03	4.000E 00	2.784E 00	9.977E-01	6.894E-03	-1.380E-02	2.993E 01			
2.546E-03	5.000E 00	3.783E 00	1.000E 00	-1.645E-05	-1.945E-05	3.000E 01			
3.055E-03	6.000E 00	4.783E 00	1.000E 00	1.647E-08	5.239E-05	3.000E 00			

TIME 0.0 SECONDS * * * STREAMWISE DIMENSION 0.1000E 01 FEET * * * *

ITERATED VALUES		DAMP		MAX LINEAR		MAX MOMENTUM	
ITS	TIME	FPPW		ERROR		ERROR	
1	0.0	0.4695	1.0000	3	9.537E-07	7	-4.801E-05
EDGE		STATIC		XI		FLUX NORM	
VELOCITY	PRESSURE	(LB/	SEC)**2	BETA	PARAMETER	ROKAP	
(FT/SEC)	(ATM)					(FT)	
3.000E 01	9.995E-01	8.536E-07	0.0		6.533E-04	1.000E 00	
WALL FR.		MOM TRANS		WALL		MASS FLUXES	
COEFF,	COEFF,			SHEAR		MECH	TOTAL
CF	RHO*UE*CF/2					REM	GAS
					(LB/SQ FT)	(LB/SQ FT SEC)	
2.673E-04	3.067E-04			2.860E-04	-0.0	0.0	
REYNOLDS		STROUHAL		MOMENTUM		SHAPE	
NUMBER		NUMBER		THICKNESS,		DISPLACE.	FACTOR,
				THETA		DELSTAR	DELSTAR/
				(FT)		(FT)	THETA
6.172E 06	0.0			2.672E-04	6.926E-04	2.592E 00	

NODAL INFORMATION

DISTANCE		ETA		STREAM		VELOCITY		SHEAR		FPPP		VELOCITY	
FROM WALL				FUNCTION		RATIO		FUNCTION				(FT/SEC)	
(FT)				(F)		(FP=U/UE)		(FPP)					
0.0	0.0			0.0		0.0		4.695E-01		6.396E-03		0.0	
4.554E-04	8.000E-01			1.495E-01		3.714E-01		4.511E-01		-5.231E-02		1.114E 01	
9.108E-04	1.600E 00			5.829E-01		6.976E-01		3.420E-01		-2.205E-01		2.093E 01	
1.366E-03	2.400E 00			1.232E 00		9.011E-01		1.676E-01		-2.154E-01		2.703E 01	
1.822E-03	3.200E 00			1.991E 00		9.801E-01		4.694E-02		-8.629E-02		2.940E 01	
2.277E-03	4.000E 00			2.784E 00		9.977E-01		6.897E-03		-1.381E-02		2.993E 01	
2.846E-03	5.000E 00			3.783E 00		1.000E 00		-1.935E-03		-2.210E-05		3.000E 01	
3.416E-03	6.000E 00			4.783E 00		1.000E 00		3.293E-08		6.087E-05		3.000E 01	

COMPUTER PROGRAM

C UNSTEADY BOUNDARY LAYER INTEGRAL PROCEDURE

C (URLIMP)

```

REAL MU
COMMON/BUICOM/BUICP,CORMA,EASE,ICORM,WDOT,I777,ISP,IX
COMMON/EDGCOM/PEM(50),PTE(50),DUES,UE(50,50),PE(50,50)
1,RHO,MU,T,DSIP(50),IDSIIP
COMMON/INTCOM/KR(20),KIN,KOUT,MATLI,MATLIJ,NETA,I,IS,NS
1,IT,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),R(8),MWE,NO
2N,ITEM,NITEM,KRI7,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/PRPCOM/TIME(50),PRE(50),PTET,S(50),ROKAP(50),RN
17SF,VKAP,NDISC,DISC(50),NSD(10),MSD(10),IPRE,RAONQ,CO
2NE,DELT,CYCLE
COMMON/PRPCOM/PR,SC,XM(5)
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
CALL ZFRCIT
1 FORMAT(A11)
2 FORMAT(F10.4)
3 FORMAT(1H1,//////////5X10HRUN TIME = F8.2,8H SECO
INDS )
DATA IAST/1H,/
DATA LAST/1H./
KIN=4
KOUT=8
R(1)=.5
R(2)=.333333333
R(3)=.166666666
R(4)=.125
R(5)=.041666666
R(6)=.033333333
R(7)=.013888888
R(8)=.003968254
RHO=.07651
MU=3.71E-07
IT=1
46 MWE=-1
EASE=1.
I777=C
IS=1
41 ITEM=1
42 CALL SETUP
43 CALL IFRAT
CALL OUTPUT
IF(NON)43,44,4C
44 ITEM=ITEM+1
DO 4C I=1,NETA
DO 4C J=1,5
FTM2(J,I)=FTM1(J,I)
FTM1(J,I)=F(J,I)
49 CONTINUE
IF(ITEM-NITEM) 42,42,45
45 IS=IS+1
IF(IS-NS) 41,41,40
40 READ(KIN,1) JAST
IF(JAST-JAST) 47,46,47
47 IF(LAST-JAST) 40,48,40
49 STOP
END

SUBROUTINE ZEROIT
COMMON/BUICOM/BUICP(8,1)
COMMON/EDGCOM/EDGCO(5,1),EDGC(5,10),EDG(1,100),ED(5,10
100)
COMMON/ERRCOM/ERRCO(5,1),ERRC(2,10),ERP(3,100)
COMMON/ETACOM/ETACO(9,1),ETAC(8,10),ETA(2,100),ET(2,10

```

```

100)
COMMON/HISCNM/HISCN(6,1),HISC(3,10),HIS(4,100)
COMMON/INTCNM/INTCN(5,1),INTC(7,10)
COMMON/NONCNM/NONCN(2,10),NONC(8,100)
COMMON/PRMCMN/PRMCO(8,1),PRMC(7,10),PRM(2,100)
COMMON/PRPCNM/PRPCN(7,1)
COMMON/TEMCNM/TEMCN(5,1),TEMC(3,10),TEM(4,100)
COMMON/VARCNM/VARCN(5,1),VARC(7,10),VAR(3,100)
COMMON/WALCNM/WALCN(3,1),WAL(1,100)
Z=0.0
DO 3 I=1,1000
DO 1 J=1,2
ET(J,I)=7
1 CONTINUE
DO 2 J=1,5
ED(J,I)=7
2 CONTINUE
3 CONTINUE
DO 8 I=1,100
EDG(1,I)=7
WAL(1,I)=7
DO 4 J=1,2
ETA(J,I)=7
PRM(J,I)=7
4 CONTINUE
DO 5 J=1,3
FRF(J,I)=7
VAR(J,I)=7
5 CONTINUE
DO 6 J=1,4
HIS(J,I)=7
TEM(J,I)=7
6 CONTINUE
DO 7 J=1,8
NON(J,I)=7
7 CONTINUE
8 CONTINUE
DO 14 I=1,10
DO 9 J=1,2
ERRC(J,I)=7
NONC(J,I)=7
9 CONTINUE
DO 10 J=1,3
HISC(J,I)=7
TEMC(J,I)=7
10 CONTINUE
DO 11 J=1,5
FDCG(J,I)=7
11 CONTINUE
DO 12 J=1,7
INTC(J,I)=7
PRMC(J,I)=7
VARC(J,I)=7
12 CONTINUE
DO 13 J=1,8
ETAC(J,I)=7
13 CONTINUE
14 CONTINUE
DO 15 J=1,3
WALCN(J,1)=7
15 CONTINUE
DO 20 J=1,6
HISCN(J,1)=7
20 CONTINUE
DO 16 J=1,5
FDCCN(J,1)=7
FRRCN(J,1)=7
INTCN(J,1)=7
TEMCN(J,1)=7
VARCN(J,1)=7
16 CONTINUE
DO 21 J=1,7

```

```

21 PRPCO(J,1)=Z
CONTINUE
DO 18 J=1,8
BUMCO(J,1)=Z
PRMCO(J,1)=Z
13 CONTINUE
DO 19 J=1,9
ETACO(J,1)=Z
10 CONTINUE
RETURN
END

SUBROUTINE SFTUP
REAL MU
DIMENSION HIST1(278),HIST2(51),HIST3(375),HIST4(100),V
1MAT(158)
COMMON/EDGCOM/PEM(50),PTE(50),DUES,UE(50,50),PE(50,50)
1,RHO,MU,T,DSIP(50),DSIP
COMMON/HISCOM/TZ,T1,T2,TF(25),C1,C2,C3,C4,BETA,ZM(5,25)
1,XI(50),HF(25,5),HUE,HHUE,DLX2,C3M(50),BETAM(50)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NITEM,KP17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/PRMCOM/TIME(50),PRE(50),PTET,S(50),ROKAP(50),RN
10SE,VKAF,NDISC,IDISC(50),NSD(10),MSD(10),IPRE,RADNO,CO
2NE,DELT,CYCLE
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
COMMON/WALCOM/FW(50),RHOVW(50),IFW,IRHOVW,JRHOVW
EQUIVALENCE (HIST1,XI),(HIST2,PTF),(HIST3,F),(HIST4,FW
1),(VMAT,TZ)
2 FORMAT(1H18X4HTIMEE12.5,56H SECCNDS - - - - -)
1 - - - - - ,6X,A5,3X,A2,1H,,A2)
3 FORMAT(1H1,3X4HTIME,E12.5,34HSECCNDS * * * STREAMWISE
1DIMENSION ,E12.5,13H FEET * * * /)
4 FORMAT(1H18X4HCASEE12.5,56H - - - - -)
1 - - - - - ,6X,A5,3X,A2,1H,,A2)
5 FORMAT(1H1,2X12H* * * CASE ,E12.5,27H * * * STREAMWI
1SE DIMENSION ,E12.5,13H FEET * * * /)
7771 FORMAT(109E10.3)
9901 FORMAT(13,7E10.3)
J=MOD(ITEM,2)+1
IF(MWE)101,154,154
101 CALL RECASE
NBT=3
NBT2=0
IS=1
IT=1
IF(KP(1)) 104,104,103
103 CALL LINMAT
104 KR17=KR(17)
154 IF(IS+ITEM-2) 153,153,1572
1572 IF(NITEM-1)1574,1262,1574
1574 WRITE(NBT)HIST1,HIST2,HIST3,HIST4,VMAT
IF(ITEM-1) 157,156,157
156 IDUM=NBT
NBT=NBT2
NBT2=IDUM
REWIND NBT
REWIND NBT2
GO TO 155
157 IF(IS-1) 153,153,155
155 READ(NBT2)HIST1,HIST2,HIST3,HIST4,VMAT
GO TO 1262
153 CONTINUE
IF(IT-1) 105,105,106
INITIAL GUESSES FOR PRINCIPAL DEPENDENT VARIABLES.
CALCULATE (KR(2)=0), INPUT (KR(2)=1), OR USE VALUES
FROM PREVIOUS CASE (KR(2)=2). NOTE: LATTER REQUIRES
SAME ETA VALUE. ITS UTILITY IS FOR REPEATED SIMILARITY
SOLUTIONS. IT OBVIOUSLY CANNOT BE USED FOR FIRST CASE.
105 CALL FIRSTG

```

```

106 CALL REFCOM
   KR(2)=2
   IS=1
1262 IF (TIME(1))1053,1054,1054
1053 TIME(1)=-TIME(1)
   WRITE(KOUT,5)TIME(ITEM),S(IS)
   TIME(1)=-TIME(1)
   GO TO 126
1054 WRITE(KOUT,3)TIME(ITEM),S(IS)
   WRITE(6,3)TIME(ITEM),S(IS)
C   COMPUTE HISTORIC INFORMATION
126 CALL HISTXI
   MWF=C
C   START OF ITERATION LOOP
158 ITS=0
   KR(17)=KR17
159 RETURN
END

SUBROUTINE REFCASE
REAL MU
COMMON/EDGCOM/FEM(50),PTF(50),DUES,UE(50,50),PE(50,50)
1,PHO,MU,T,OSIP(50),IDISIP
COMMON/ETACOM/ETA(25),NETA(24),DSQ(24),DCU(24),R1(24),
1R2(24),LAR(100),BAL(73,28)
COMMON/INTCOM/KP(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAM,NIFO,NNLEO,NRNLI,ITS,CASE(15),R(3),MWF,NO
2N,ITEM,NITEM,KP17,NBT,NBT2,IDENT,UMES,FREQ,EPS
COMMON/PRMCOM/TIME(50),PRE(50),PTET,S(50),ROKAP(50),RN
10SF,VKAP,NDISC,IDISC(50),NSQ(10),MSQ(10),IPRE,PADNO,CO
2NE,DELT,CYCLE
COMMON/PRPCOM/PP,SC,XM(5)
COMMON/WALCOM/FW(50),PHOVW(50),IFW,IRHGVW,JPHGVW
1 FORMAT(1H1/////////15X40HUNSTEADY BOUNDARY LAYER INTEGRA
1L MATRIX PROCEDURE //
127X25HDEPARTMENT OF AERONAUTICS /27X25HNAVAL POSTGRADU
24TE SCHOOL /30X20HMONTEREY, CALIFORNIA //32X16HINVESTI
3GATION OF //10X15A4, //32X15
4HCONTROL NUMBERS //11X56H1 2 3 4 5 6 7 8 9 10 1
51 12 13 14 15 16 17 18 19 20 /9X2013//35X10HETA VALUES
6 /4(5X1F7E10.3//))
2 FORMAT(20X17HNOSE RADIUS = F10.5,13H FT )
3 FORMAT(20X17HCONE HALF ANGLE = F10.5,13H DEG )
4 FORMAT(20X17HDENSITY = F10.9,13H LB/FT CU )
5 FORMAT(20X17HVISCOSITY = 1PE10.3,13H LB SEC/FT S
10 )
6 FORMAT(20X17HTIME GRID = 1PE10.3,13H SEC
1 )
7 FORMAT(20X17HCYCLE TIME = 1PE10.3,13H SEC
1 )
9 FORMAT(1H1//////////31X18HSTATION INPUT DATA //1X11HDI
1STANCE,FT ,12X5(1X1PE10.3)/3(24X5(1X1PE10.3//))
9 FORMAT(1X54ROCKAP,18X5(1X1PE10.3)/9(24X5(1X1PE10.3//))
10 FORMAT(1X20HPRESSURE RATIO, MEAN,3X/9(24X5(1X1PE10.3)
15(1X1PE10.3//))
11 FORMAT(5X48HTEMPERATURE TOTAL MEAN
1 OSCIL OSCIL /21X53HPRESSURE VELO
2CITY AMPLITUDE FREQUENCY /7X66H(DEG R) (A
3TM) (FT/SEC) PARAMETER (HERTZ) /5(5X
41PE10.3//))
12 FORMAT(1X21HENTROPY DROP,BTU/LB R ,2)/9(24X5(1X1PE10.3
1X5(1X1PE10.3//))
13 FORMAT(1X22HMASS FLUX,LB/SEC FT**2 ,1X5(1X1PE10.3)/9(2
14X5(1X1PE10.3//))
   IF (ITEM-1)302,302,194
302 CALL INPUT
   WRITE(KOUT,1)CASE,KR,(FTA(JN),JN=1,NETA)
   WRITE(KOUT,11)T,PTET,UMES,EPS,FREQ
   NITEM=NTIME
   NTIME=1
C   COMPUTE INFORMATION NEEDED TO CONSIDER DISCONTINUITIES

```

```

303 J=1
102 MSD(1)=1
    IDISC(1)=0
    S(1)=ABS(S(1))
    IF (NS-1) 105,105,1021
1021 DO 101 IS=2,NS
    IDISC(IS)=0
    IF(S(IS)) 103,101,101
    103 NSD(J)=IS-MSD(J)+1
    S(IS)=-S(IS)
    IDISC(IS)=1
    J=J+1
    MSD(J)=IS
101 CONTINUE
105 NSD(J)=NS-MSD(J)+1
    NDISC=J-1
    IF(IABS(KR(6)-2)-1)207,208,207
207 IF(NS-1)234,234,2071
2071 IIS=2
    LN7=1
    IF(ROKAP(1)) 223,226,226
223 RADNO=-ROKAP(1)
    ROKAP(1)=0.
    DO 229 IS=2,NS
    IF(ROKAP(IS)) 224,224,225
224 IF(KR(6)) 221,221,222
222 ROKAP(IS)=S(IS)*SIN(RADNO/57.29578)
    GO TO 229
221 ROKAP(IS)=RADNO*SIN(S(IS)/RADNO)
229 CONTINUE
    GO TO 234
225 IF (IS-NS) 2251,234,234
2251 IIS=IS+1
    LN7=IS
226 DO 233 IS=IIS,NS
    IF(ROKAP(IS)) 233,233,227
227 IF(IS-1-LN7) 232,232,228
228 LN7=LN7+1
    ROKAP(LN7)=ROKAP(LN7-1)+(S(LN7)-S(LN7-1))/(S(IS)-S(LN7-1))
    GO TO 227
232 LN7=IS
233 CONTINUE
234 VKAP=1.
    GO TO 210
209 DO 209 IS=1,NS
209 ROKAP(IS)=1.
    VKAP=0.
210 CONTINUE
181 STEFF = .481E-12
    IF(KR(6)-1)182,182,183
182 WRITE(KCUT,2)RADNO
    WRITE(KCUT,3)CONE
183 WRITE(KCUT,4)RHO
    WRITE(KCUT,5)MU
    WRITE(KCUT,6)DELT
    WRITE(KCUT,7)CYCLE
    WRITE(KCUT,8)(S(I),I=1,NS)
    WRITE(KCUT,9)(ROKAP(I),I=1,NS)
    WRITE(KCUT,10)(PPF(I),I=1,NS)
    WRITE(KCUT,12)(DSIP(I),I=1,NS)
    WRITE(KCUT,13)(RHOVW(I),I=1,NS)
184 RETURN
END

SUBROUTINE INPUT
REAL MU
COMMON/EDGCOM/PEM(50),PTE(50),DUES,UE(50,50),PF(50,50)
1,RHO,MU,T,DSIP(50),IDSIP
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
1B2(24),LAR(100),BA1(73,28)

```

```

COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NTFM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/PRMCOM/TIME(50),PRE(50),PTET,C(50),ROKAP(50),PN
10SE,VKAP,NDISC,DISC(50),NSD(10),MSD(10),IPRE,RADNO,CO
2NE,DELT,CYCLE
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
COMMON/WALCOM/FW(50),RHOVW(50),IFW,IRHOVW,JRHOVW
1 FORMAT(20I1,15A4)
2 FORMAT(2I3,E10.4)
3 FORMAT(8E10.4)
4 FORMAT(5(2X5I4.7))
READ (KIN,1)KR,CASE
READ (KIN,2)NTIME
READ (KIN,2)NS
READ (KIN,3)(S(15),IS=1,NS)
IF (KR(1))103,103,100
100 IF(KR(12))101,101,102
101 READ (KIN,2)NETA
READ (KIN,3)(ETA(I),I=1,NETA)
GO TO 103
102 CALL STCETA
103 IF (KR(6)-1)104,104,105
104 READ (KIN,3)CONE,RNOSE
105 IF(ABS(KR(6)-2)-1)106,107,106
106 READ (KIN,3)(ROKAP(15),IS=1,NS)
107 READ (KIN,3)PTET
1071 IF (KR(2)-1)112,110,112
110 READ (KIN,4)((F(I,J),J=1,NETA),I=1,4)
112 IF(ITEM-1)115,113,115
113 IDISP=1
IDISP=1
DO 114 I=1,NS
114 DSI(I)=C.
115 IF(KR(5)-5)118,116,118
116 IF(IDISP-ITEM)118,117,118
117 READ (KIN,3)(DISP(I),I=1,NS)
READ (KIN,2)IDISP
118 IF (IDISP-ITEM)120,119,120
119 READ (KIN,4)(PRE(I),I=1,NS)
READ (KIN,2)IPRE
120 JRHOVW=C
IF(ITEM-1)121,121,122
121 IFW=1
IRHOVW=1
122 READ (KIN,3)T
122 IF(ITEM-1)130,130,131
130 IF(KR(9)-1)131,135,138
131 IF(IFW-ITEM)132,133,132
132 IF(IFW-1)134,135,134
133 READ (KIN,3)(FW(I),I=1,NS)
READ (KIN,2)IFW
134 GO TO 138
135 IF(IRHOVW-ITEM)136,137,136
136 GO TO 138
137 READ (KIN,3)(RHOVW(I),I=1,NS)
READ (KIN,2)IRHOVW
138 READ(KIN,3)UMFS,FREQ,EPS
JRHOVW=C
IF (FREQ)141,141,139
139 CYCLE=1./FREQ
RAG=FLOAT(NTIME)
DELT=CYCLE/RAG
TIME(1)=0.
DO 140 I=2,NTIME
140 TIME(I)=TIME(I-1)+DELT
GO TO 142
141 TIME(1)=C.
DELT=0.
CYCLE=0.
142 CONTINUE
IF(KR(16))143,145,143

```

```

143 DD 144 I=2,NS
144 PRF(I)=PRE(I)*RHO*UMFS*UMFS/136175.57+1.
145 RETURN
END

```

```

SURROUTINE STDSTA
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
1B2(24),LAR(100),BA1(73,28)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAW,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NITEM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
IF(KR(12)-2)4,5,1
1 IF(KR(12)-4)6,7,2
2 IF(KR(12)-6)8,9,3
3 IF(KR(12)-8)10,11,12
4 NETA=8
ETA(1)=C.0
ETA(2)=C.5
ETA(3)=1.0
ETA(4)=1.5
ETA(5)=2.0
ETA(6)=3.0
ETA(7)=5.0
ETA(8)=6.0
GO TO 13
5 NETA=9
ETA( 1)=0.00
ETA( 2)=C.80
ETA( 3)=1.60
ETA( 4)=2.40
ETA( 5)=3.20
ETA( 6)=4.00
ETA( 7)=5.00
ETA( 8)=6.0
GO TO 13
6 NETA=10
ETA( 1)=C.00
ETA( 2)=0.60
ETA( 3)=1.20
ETA( 4)=1.80
ETA( 5)=2.40
ETA( 6)=3.00
ETA( 7)=3.60
ETA( 8)=4.20
ETA( 9)=5.00
ETA(10)=6.00
GO TO 13
7 NETA=12
ETA( 1)=0.00
ETA( 2)=0.50
ETA( 3)=1.00
ETA( 4)=1.50
ETA( 5)=2.00
ETA( 6)=2.50
ETA( 7)=3.00
ETA( 8)=3.50
ETA( 9)=4.00
ETA(10)=4.50
ETA(11)=5.00
ETA(12)=6.00
GO TO 13
8 NETA=14
ETA( 1)=0.00
ETA( 2)=0.40
ETA( 3)=0.80
ETA( 4)=1.20
ETA( 5)=1.60
ETA( 6)=2.00
ETA( 7)=2.40
ETA( 8)=2.80
ETA( 9)=3.20

```

```

ETA(10)=3.60
ETA(11)=4.00
ETA(12)=4.50
ETA(13)=5.00
ETA(14)=6.00
GO TO 13
9  NETA=15
ETA(1)=0.00
ETA(2)=0.20
ETA(3)=0.40
ETA(4)=0.60
ETA(5)=0.80
ETA(6)=1.00
ETA(7)=1.20
ETA(8)=1.40
ETA(9)=1.60
ETA(10)=2.00
ETA(11)=2.50
ETA(12)=3.25
ETA(13)=4.00
ETA(14)=5.00
ETA(15)=6.00
GO TO 13
10 NETA=18
ETA(1)=0.00
ETA(2)=0.30
ETA(3)=0.60
ETA(4)=0.90
ETA(5)=1.20
ETA(6)=1.50
ETA(7)=1.80
ETA(8)=2.10
ETA(9)=2.40
ETA(10)=2.70
ETA(11)=3.00
ETA(12)=3.30
ETA(13)=3.60
ETA(14)=3.90
ETA(15)=4.20
ETA(16)=4.60
ETA(17)=5.00
ETA(18)=6.00
GO TO 13
11 NETA=22
ETA(1)=0.000
ETA(2)=0.125
ETA(3)=0.250
ETA(4)=0.375
ETA(5)=0.500
ETA(6)=0.625
ETA(7)=0.750
ETA(8)=0.875
ETA(9)=1.000
ETA(10)=1.125
ETA(11)=1.250
ETA(12)=1.375
ETA(13)=1.500
ETA(14)=1.750
ETA(15)=2.000
ETA(16)=2.250
ETA(17)=2.500
ETA(18)=3.000
ETA(19)=3.500
ETA(20)=4.000
ETA(21)=5.000
ETA(22)=6.000
GO TO 13
12 NETA=25
ETA(1)=0.00
ETA(2)=0.20
ETA(3)=0.40
ETA(4)=0.60

```

```

ETA( 5)=0.80
ETA( 6)=1.00
ETA( 7)=1.20
ETA( 8)=1.40
ETA( 9)=1.40
ETA(10)=1.80
ETA(11)=2.00
ETA(12)=2.20
ETA(13)=2.40
ETA(14)=2.60
ETA(15)=2.80
ETA(16)=3.00
ETA(17)=3.20
ETA(18)=3.40
ETA(19)=3.60
ETA(20)=3.80
ETA(21)=4.00
ETA(22)=4.25
ETA(23)=4.50
ETA(24)=5.00
ETA(25)=6.00
13 RETURN
END

SUBROUTINE LINMAT
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
1B2(24),LAR(100),BA1(73,28)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,I,T,NTIME,NAW,NLEQ,NNLEQ,NRNL,IIS,CASE(15),B(8),MWE,NO
2N,IITEM,NITEM,KR17,NRT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/TEMCOM/SPDUM2(10),DER(50),DUMM1(25),SLOPE(25),R
1EDUM(25),SDUM1(50),SDUM2(50),FWDUM(50),XICON(50),FWCO
2N(50),DUDS(50)
DO 104 I=2,NETA
DETA(I-1)=ETA(I)-ETA(I-1)
DSQ(I-1)=DETA(I-1)*DETA(I-1)
B1(I-1)=B(3)*DSQ(I-1)
B2(I-1)=2.*B1(I-1)
104 DCU(I-1)=DETA(I-1)*DSQ(I-1)
MAT1I=3*NETA-2
MAT1J=NETA+2
DO 105 I=1,MAT1I
DO 105 J=1,MAT1J
105 BA1(1,J)=0.
BA1(1,1)=1.
BA1(MAT1I,2)=-1.
DO 106 I=2,NETA
IF (KR(10)) 1041,1041,1042
1042 IF (I-NETA) 1044,1041,1044
1041 DUM1=B(5)
DUM2=B(4)
DUM3=B(3)
DUM4=B(2)
DUM5=B(1)
DUM6=B(1)
GO TO 1043
1044 DUM1=0.
DUM2=B(3)
DUM3=0.
DUM4=B(1)
DUM5=0.
DUM6=1.
1043 BA1(I-1,I+1)=DUM2*DCU(I-1)
BA1(I-1,I+2)=DUM1*DCU(I-1)
J=I+NETA-1
BA1(J,I+1)=DUM4*DSQ(I-1)
BA1(J,I+2)=DUM3*DSQ(I-1)
J=J+NETA-1
BA1(J,I+1)=DUM6*DETA(I-1)
106 BA1(J,I+2)=DUM5*DETA(I-1)
9060 FORMAT(2X1P12E10.3)

```

```

9064 FORMAT(2X37HLINEAR MATRIX FOR MOMENTUM EQUATIONS,I3,2H
1 X13,27H, BEFOFE AND AFTER SOLUTION)
IF(KR(15)-1) 9062,9062,9061
9061 WRITE(KOUT,9064) MAT1I,MAT1J
DO 9063 I=1,MAT1I
9063 WRITE(KOUT,9060)(RA1(I,J),J=1,MAT1J)
9062 DO 107 LI=1,MAT1J
107 CALL MATONE(RA1(I,LI))
IF (KR(15)-1)9072,9072,9065
9066 DO 9067 I=1,MAT1I
9067 WRITE(KOUT,9060)(RA1(I,J),J=1,MAT1J)
9072 NLFQ=MAT1I
NNLEQ=MAT1J
NAM=NNLFQ-1
NRNL=1
9002 CONTINUE
RETURN
END

```

```

SUBROUTINE MATONE(E)
DIMENSION E(1)
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
1B2(24),LAR(100),BA1(73,28)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,I,T,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,IITEM,NITEM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,FPS
I=MAT1I
DO 201 J=3,NETA
I=I-1
201 E(I)=E(I)+E(I+1)
I=NETA
M=NETA+NETA-1
K=C
DO 202 J=2,NETA
I=I+1
M=M+1
K=K+1
202 E(I)=E(I-1)-E(I)+DETA(K)*E(M)
203 E(I)=-F(1)+DETA(1)*E(K+1)+DSQ(1)/2.*E(I+1)
I=I+1
K=K+1
M=M+1
DO 204 J=3,NETA
I=I+1
K=K+1
M=M+1
204 E(M)=E(M-1)-E(M)+DETA(M)*E(K)+DSQ(M)/2.*E(I)
RETURN
END

```

```

SUBROUTINE PERAY(N,C,D,NNN,IS,ND)
DIMENSION D(ND,1),SD(20),C(ND,1),L(20),S(28),LL(28),LL
1L(28)
COMMON/INTCOM/KR(20),KIN,KOUT
C DIRECT INVERSION PROCEDURE -- C IS REPLACED BY C**=1
N1=N+1
DO 11 I=1,N
LL(I)=I
113 L(I)=I
11 CONTINUE
IX=-1
IF(IS+2) 111,109,111
106 FORMAT(11H L(I),I=1,I3,5X (30I3))
107 FORMAT(15H ((C(I,J),J=1,I3,12H),(D(J),J=1,I3,6H),I=1
1,I3,15H) BEFORE PERAY)
108 FORMAT(2X1P11E10.3/(12X1P10F10.3))
109 WRITE(KOUT,107) N,NNN,N
WRITE(KOUT,106)N,(L(I),I=1,N)
IX=0
DO 110 I=1,N

```

```

110 WRITE(KOUT,108) (C(I,J),J=1,N),(D(I,J),J=1,NNN)
111 IS=-1
C   TRIANGULATE MATRIX
    DO 15 I=1,N
    DO 160 M=1,N
160 S(M)=ABS(C(I,M))
    IF(IS) 18,16,16
    18 IS=0
    GO TO 12
C   REDUCE ROW I BY PRECEEDING ROWS
    DO 17 J=2,I
    K=L(J-1)
    DIV=-C(I,K)
    IF(DIV) 161,17,161
161 C(I,K)=0.
    DO 162 M=1,N
    DIVC=DIV*C(J-1,M)
    S(M)=AMAX1(S(M),ABS(DIVC))
162 C(I,M)=C(I,M)+DIVC
    IF(NNN) 17,17,163
163 DO 164 M=1,NNN
164 D(I,M)=D(I,M)+DIV*D(J-1,M)
    17 CONTINUE
C   SEEK MAXIMUM PIVOT
    12 DIV=0.
    DO 13 JJ=I,N
    M=L(JJ)
    IF(ABS(C(I,M))-DIV) 13,13,121
121 DIV=ABS(C(I,M))
    K=M
    J=JJ
    13 CONTINUE
    SD(I)=DIV/S(K)
    L(J)=L(I)
    L(I)=K
    IF(SD(I)-1.F-8) 130,130,14
130 C(I,K)=0.
    IF(SD(I)) 131,131,12
C   SINGULAR MATRIX RETURN
131 IS=-1
    WRITE(KOUT,151) (I,L(I),SD(I),I=1,I)
    RETURN
    14 DIV=C(I,K)
    C(I,K)=1.0
    K=LLL(J)
    LLL(J)=LLL(I)
    LLL(I)=K
    LL(K)=I
C   NORMALIZE ROW
    IF(NNN) 143,143,141
141 DO 142 J=1,NNN
142 D(I,J)=D(I,J)/DIV
143 DO 15 J=1,N
    15 C(I,J)=C(I,J)/DIV
    IF(IX) 152,150,152
151 FORMAT(24H PIVOT ROW/COL/RES.RATIO 5(I4,1H/I3,1H/E9.2,
11H,))
150 WRITE(KOUT,151) (I,L(I),SD(I),I=1,N)
C   DIAGONALIZE MATRIX
152 NM=N-1
    DO 20 I=1,NM
    K=L(I+1)
    DO 20 J=1,I
    DIV=-C(J,K)
    IF(DIV) 19,20,19
    19 C(J,K)=0.
    IF(NNN) 191,191,192
192 DO 193 M=1,NNN
193 D(J,M)=D(J,M)+DIV*D(I+1,M)
191 DO 201 M=1,N
201 C(J,M)=C(J,M)+DIV*C(I+1,M)
    20 CONTINUE

```

```

C      INTERCHANGE COLUMNS
      DO 30 II=1,N
      I=II
21     J=L(I)
      L(I)=I
      IF(J-I)22,30,22
22     IF(I<J)25,23,25
23     DO 24 M=1,N
      S(M)=C(M,I)
24     C(M,I)=C(M,J)
      IS=I
      I=J
      GO TO 21
25     IF(I<J)26,28,26
26     DO 27 M=1,N
27     C(M,I)=C(M,J)
      I=J
      GO TO 21
28     DO 29 M=1,N
29     C(M,I)=S(M)
      IS=0
30     CONTINUE
C      INTERCHANGE ROWS
      DO 40 II=1,N
      I=II
31     J=LL(I)
      LL(I)=I
      IF(J-I)32,40,32
32     IF(I<J)35,33,35
33     DO 34 M=1,N
      S(M)=C(I,M)
34     C(I,M)=C(J,M)
      IF(NNN) 343,343,341
341    DO 342 M=1,NNN
342    S(M)=D(I,M)
343    D(I,M)=D(J,M)
      IS=I
      I=J
      GO TO 31
35     IF(I<J)36,38,36
36     DO 37 M=1,N
37     C(I,M)=C(J,M)
      IF(NNN) 373,373,371
371    DO 372 M=1,NNN
372    D(I,M)=D(J,M)
373    I=J
      GO TO 31
38     DO 39 M=1,N
39     C(I,M)=S(M)
      IF(NNN) 393,393,391
391    DO 392 M=1,NNN
392    D(I,M)=S(M)
393    IS=0
40     CONTINUE
      IF(JX) 411,409,411
407    FORMAT(15H ((C(I,J),J=1,I3,12H), (D(J),J=1,I3, 6H), I=1
1,I3,15H) AFTER REPAY )
409    WRITE(KOUT,407) N,NNN,N
      DO 410 I=1,N
410    WRITE(KOUT,108) (C(I,J),J=1,N), (D(I,J),J=1,NNN)
411    RETURN
      END

```

```

SUBROUTINE FIRSTG
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),R1(24),
1B2(24),LAR(100),RA1(73,28)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,I,T,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,I,TEM,NTIME,KP17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/PPMCOM/TIME(50),PRE(50),PTET,S(50),ROKAP(50),RN
10SE,VKAP,NDISC,IDISC(50),NSD(10),MSD(10),IPRE,RADNO,CO

```

```

2NF,DELT,CYCLE
COMMON/VARCON/F(5,25),FTM1(5,25),FTM2(5,25)
IF(KR(2)-1) 112,152,152
112 DO 128 I=1,NETA
    F(1,I)=0.
    DUM1=ETA(I)-4.
    IF(DUM1) 115,116,116
115 DUM1=-C.1*DUM1
    DUM2=0.25*ETA(I)
    F(2,I)=DUM2
    F(3,I)=DUM1
    GO TO 114
116 F(2,I)=1.0
    F(3,I)=0.
114 F(4,I)=0.
128 CONTINUE
    IF(KR(20)) 152,152,200
200 DO 208 I=1,NETA
    IF(ETA(I)-5.) 201,201,202
201 F(1,I)=ETA(I)
    F(2,I)=ETA(I)/5.
    F(3,I)=.1*(5.-ETA(I))
    GO TO 203
202 F(1,I)=F.
    F(2,I)=1.
    F(3,I)=0.
203 IF(ETA(I)-2.) 204,204,205
204 F(4,I)=-.1*ETA(I)
    GO TO 208
205 IF(ETA(I)-5.) 206,206,207
206 F(4,I)=-.067*(5.-ETA(I))
    GO TO 208
207 F(4,I)=0.
208 CONTINUE
    F(1,1)=0.
    F(2,1)=C.
    F(2,NETA)=1.
    F(3,NETA)=0.
    F(4,NETA)=0.
152 CONTINUE
9901 FORMAT(4(2X7E10.3/))
9904 FORMAT(2X19HDEBUG FIRST GUESSES)
IF(KR(15)) 9902,9903,9902
9902 CONTINUE
WRITE(KOUT,9904)
WRITE(KOUT,9901)((F(I,J),J=1,NETA),I=1,4)
9903 CONTINUE
RETURN
END

```

```

SUBROUTINE REFCON
REAL MU
COMMON/EDGCON/PEM(50),PTE(50),DUES,UE(50,50),PE(50,50)
1,RHO,MU,T,DSIP(50),IDSIP
COMMON/ETACON/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
192(24),LAR(100),BAL(73,28)
COMMON/HISCON/T2,T1,T2,TF(25),C1,C2,C3,C4,BETA,ZM(5,25)
1,XI(50),HF(25,5),HUE,HHUE,DLX2,C3M(50),BETAM(50)
COMMON/INTCON/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NITEM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/PRMCON/TIME(50),PRE(50),PTET,S(50),ROKAP(50),RN
10SE,VKAP,NDISC,IDISC(50),NSD(10),MSD(10),IPRE,RADNO,CO
2NE,DELT,CYCLE
COMMON/TEMCON/SPDUM2(10),DER(50),DUMM1(25),SLOPE(25),R
1EDUM(25),SDUM1(50),SDUM2(50),FWDUM(50),XICON(50),FWCO
2N(50),DUDS(50)
COMMON/WALCON/FW(50),RHOVW(50),IFW,IRHOVW,JRHOVW
1 FORMAT(1H1//////////29X21HSTATION COMPUTED DATA /30X5H
1TIME=IPE10.3,4H SEC //)
2 FORMAT(15X68HSTREAMWISE XI BETA

```

```

1          FDGE          STATIC /5X69HDIMENSION          (LB/
2          VELOCITY          PRESSURE /7X6
3          35H(FEET)          SEC)**2          (FT/SEC)
4          (ATM) /)
3          FORMAT(5(5X1PE10.3))
9901          FORMAT(5(5X1PE10.3))
1532          IF(ITEM-1)605,1538,605
1538          IDSIP=1
          I2PRF=1
          DO 1540 I=1,NS
1540          DSIP(I)=C.
1534          IF (KR(6)-3)1535,1541,1535
1535          IF (PRF(I))1541,1542,1541
1542          DO 1543 I=2,NS
          IF(PRF(I)) 1544,1543,1544
1544          L=I
          GO TO 1545
1543          CONTINUE
1545          RNOSE=S(L)/SQRT(1.-PRE(L))
          DO 1546 I=2,L
1546          PRE(I-1)=1.-S(I-1)/RNOSE*S(I-1)/RNOSE
1541          DO 1531 I=1,NS
          PTE(I)=PTET
1531          PEM(I)=PTET*PRE(I)
9002          PI=3.14159
          DO 8 I=1,NS
          DO 8 II=1,NITEM
          DPDS=(PTET-PEM(I))/RHC
          DPDS=136175.57*DPDS
          UE(I,II)=SQRT(DPDS)
          UE(I,II)=UE(I,II)*(1.+EPS*COS((FREQ*S(I)/UE(I,II))+
1          FLOAT(II)*2.*PI/FLOAT(NITEM)))
          PE(I,II)=PTET-(RHC*UE(I,II)**2./136175.57)
9          CONTINUE
205          IF(KP(15)) 9903,9904,9903
9903          CONTINUE
          WRITE(KOUT,9901)UE(15,ITEM),PTE(15),T,RHC,MU
9904          CONTINUE
C          FROCES DUDS TO ZERO FOR FLAT PLATE TEST.
C          IF (KR(7))81,83,81
81          DO 82 JS=2,NS
82          UE(JS,ITEM)=UE(JS-1,ITEM)
93          CONTINUE
C          END OF EDGE PROPERTY LOOP, START OF BETA AND XI
C          CALCULATION.
605          XI(1)=C.
          J=NDISC+1
          DO 111 II=1,J
          K=NSD(II)
          M=MSD(II)
          MM=M+1
          LI=K+M-1
          IF (II-1) 6052,6052, 402
6052          IF (KP(6)-1) 400,401,402
C          AXISYMETRIC PLUNT
400          DO 403 I=M,LL
          SDUM2(I)=S(I)*S(I)
          IF(S(I)) 403,403,4031
4031          DUM1=RKAP(I)/S(I)
          XICON(II)=UE(I,ITEM)/S(I)*RHO/4.*MU*DUM1*DU41
403          SDUM1(I)=SDUM2(I)*SDUM2(I)
          OT=4
          BETAM(1)=0.5
          GO TO 406
C          PLANAR PLUNT
401          DO 404 I=M,LL
          SDUM2(I)=S(I)
          IF(S(I)) 404,404,4041
4041          XICON(II)=UE(I,ITEM)/S(I)*RHO/2.*MU
404          SDUM1(I)=S(I)*S(I)
          OT=2
          BETAM(1)=1.

```

```

GO TO 406
402 IF (KR(6)-2) 408,407,408
C AXISYMETRIC SHARP
407 DO 400 I=M,LL
SDUM2(I)=S(I)*S(I)
DUM1=ROKAP(I)/S(I)
XICON(I)=RHO*UE(I,ITEM)*MU*DUM1/3.*DUM1
409 SDUM1(I)=S(I)*SDUM2(I)
XI(1) =XICON(I)*S(1)*S(1)
QI=3.
IF (II-1) 4051,4051,406
C PLANAR SHARP
408 DO 405 I=M,LL
SDUM2(I)=S(I)
405 SDUM1(I)=RHO*UE(I,ITEM)*MU*ROKAP(I)*ROKAP(I)
QI=1.
IF (II-1) 4052,4052,406
4052 XI(1) =XICON(I)*S(1)
4051 MM=M
FW(1)=0.
406 CONTINUE
C CALL SLOPQ(K,S(M),UE(M,ITEM),DUDS(M),DER(M))
FORCES DUDS TO ZERO FOR FLAT PLATE TEST.
IF (KR(7)) 84,86,84
R4 DO 85 JS=1,NS
R5 DUDS(JS)=0.0
R6 CONTINUE
IF (KR(6)-2) 4066,4062,4062
4066 IF (M-1) 4061,4061,4062
4061 IF (RNOSE) 4063,4063,4064
C DUES COMPUTED BY SLOPQ
4063 DUES=DUDS(1)
GO TO 4065
C DUES FROM EFFECTIVE NOSE RADIUS USING NEWTONIAN FLOW
4064 DUES=SQRT(2./RHO*FE(1,ITEM)*32.17405*2116.2)/RNOSE
4065 XICON(1)=RHO*MU/(2.*VKAP+2.)*DUES
4062 CALL SLOPQ(K,SDUM1(M),XICON(M),DER(M),XI(M))
IF (LL-MM) 111,4101,4101
4101 DO 410 L=MM,LL
410 BFTAM(L)=2./QI*XI(L)/UE(L,ITEM)*S(L)/SDUM1(L)*DUDS(L)/
XICON(L)
111 CONTINUE
9803 FPMAT(8E10.4)
C CALCULATION OF C3 MATRIX
DO 138 I=1,NS
IF (KR(6)-1) 137,137,158
137 IF (I-1) 139,139,158
139 C3M(I)=-SQRT(BFTAM(I)/(DUES*RHO*MI))
GO TO 138
158 C3M(I)=-SQRT(2.*XI(I)/(RHO*ROKAP(I)*UE(I,ITEM)*MU))
138 CONTINUE
JPHOVW=1
7043 IF (JPHOVW) 7047,7047,7045
C CALCULATE FW IF RHOVW GIVEN
7045 J=NDISC+1
DO 7046 II=1,J
K=NSD(II)
M=MSD(II)
LL=K+M-1
DO 209 I=M,LL
IF (KR(8)) 7049,7049,2291
7049 RHOVW(I)=RHOVW(I)*C3M(I)
2291 IF (II-1) 7048,7048,230
7048 IF (KR(6)) 229,229,230
C VALID AT AXISYMETRIC STAGNATION POINT ONLY
229 FWCON(I)=-RHOVW(I)/(2.*C3M(I))
IF (I-1) 209,209,232
C MODIFICATION FOR AXISYMETRIC BLUNT AWAY FROM
C STAGNATION POINT.
232 FWCON(I)=FWCON(I)/S(I)*ROKAP(I)
GO TO 209

```

```

C      VALID FOR ALL PLANAR
230  FWCON(I)=-RHOVW(I)/C3M(I)
      IF(KR(4)-2) 209,236,209
C      MODIFICATION FOR AXISYMETRIC SHARP
236  FWCON(I)=FWCON(I)/S(I)*ROKAP(I)/2.
209  CONTINUE
      FWDUM(1)=FWCON(1)*S(1)
      IF(KR(4)-2) 241,237,241
C      MODIFICATION FOR AXISYMETRIC SHARP
237  FWDUM(1)=FWDUM(1)*S(1)
241  CONTINUE
7046  CALL SLOPQ(K,SDUM2(M),FWCON(M),DER(M),FWDUM(M))
      DO 126 I=1,NS
      IF(I-1) 124,124,123
124  IF (KR(6)-1) 133,133,123
133  IF (S(I)) 113,113,123
113  FW(I)=RHOVW(I)
      GO TO 126
123  FW(I)=FWDUM(I)/SQRT(2.*XI(I))
126  CONTINUE
7047  WRITE(KOUT,1)TIME(ITEM)
      WRITE(KOUT,2)
      DO 7050 I=1,NS
7050  WRITE(KOUT,3)S(I),XI(I),BETAM(I),UE(I,ITEM),PE(I,ITEM)
      RETURN
      END

```

```

SUBROUTINE SLOPQ(N,X,Y,S,Z)
  DIMENSION X(1),Y(1),S(1),Z(1)
  IF(N-1) 5,6,8
  8  S(2)=(Y(2)-Y(1))/(X(2)-X(1))
  S(1)=S(2)
  QC=S(2)
  DO 7 I=1,N
  IF(I+1-N) 2,1,6
  1  QB=QC
  IF (I-2) 7,6,5
  2  XQT=X(I)-X(I+1)
  XTT=X(I+1)-X(I+2)
  XTC=X(I+2)-X(I)
  AA=Y(I)/(XCT*XTC)
  XQTT=XQT*XTT
  AB=Y(I+1)/XQTT
  AC=Y(I+2)/(XTT*XTC)
  AAA=AA*XTT
  ABB=AB*XTC
  ACC=AC*XCT
  QA=QC
  QB=S(I)
  QC=S(I+1)
  S(I)=AA*(XTC-XQT)+ABB-ACC
  S(I+1)=AB*(XCT-XTT)+ACC-AAA
  S(I+2)=AC*(XTT-XTC)+AAA-ABB
  3  IF(I-2) 7,5,4
  4  S(I)=(S(I)+QA)/2.
  5  S(I)=(S(I)+QB)/2.
  6  XD=X(I)-X(I-1)
  YS=Y(I)+Y(I-1)
  SD=S(I)-S(I-1)
  SS=S(I)
  Z(I)=Z(I-1)+XD/2.*(YS-XD/6.*SD)
  S(I)=SS
  7  CONTINUE
  9  RETURN
  END

```

```

SUBROUTINE HISTXI
  REAL MU
  DIMENSION FD(4)
  COMMON/EDGCOM/PEM(50),PTE(50),DUES,UE(50,50),PE(50,50)

```

```

1,RHO,MU,T,DSIP(50),IDSTP
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
1B2(24),LAR(100),BA1(73,28)
COMMON/HISCOM/TZ,T1,T2,TF(25),C1,C2,C3,C4,BETA,ZM(5,25
1),XI(50),HF(25,5),HUF,HHUE,DLX2,C3M(50),BETAM(50)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,I,T,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NITEM,KR17,NBT,NBT2,IDENT,IJMS,FREQ,EPS
COMMON/PRMCOM/TIME(50),PRE(50),PTET,S(50),ROKAP(50),RN
10SF,VKAP,NDISC,DISC(50),NSD(10),MSD(10),IPRE,RADNO,CO
2NF,DELT,CYCLE
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
C INITIALIZE AXIAL VARIATION TERMS
NUL=0
DLX1=0.
DO 120 I=1,4
120 FD(I)=0.
IF (KR(6)-1) 400,400,401
400 IF (IS-3) 154,153,153
401 IF (IS-2) 154,153,153
153 IF (KR(3)-1) 154,143,143
143 DLX1=ALOG(XI(IS)/XI(IS-1))
IF (KR(6)-1) 402,402,403
402 IF (IS-3) 154,155,157
403 IF (IS-2) 154,155,157
157 IF (KR(3)-1) 154,155,144
154 D7=0.
D1=C.
D2=0.
M=NETA-1
DO 140 I=1,4
DO 140 J=1,M
7M(I,J)=0.0
140 CONTINUE
DO 141 I=1,NETA
DO 142 J=1,3
HF(I,J)=C.
142 CONTINUE
HF(I,4)=0.
141 HF(I,5)=0.
DLX2=0.
GO TO 130
C COMPUTE TWO- OR THREE-POINT DIFFERENCE RELATIONS
155 D7=2./DLX1
D1=-D7
D2=0.
GO TO 145
144 J=DISC(IS-1)
IF(J)121,121,155
121 DZ=DLX1+DLX2
D1=-DZ/(DLX1*DLX2)*2.
D2=DLX1/(DZ*DLX2)*2.
DZ=-D1-D2
145 DLX2=DLX1
122 FD(3)=D1*F(4,1)+D2*HF(1,4)
DO 147 I=2,NETA
FD(1)=D1*F(2,I)+D2*HF(I,2)
FD(2)=D1*F(3,I)+D2*HF(I,3)
FD(4)=FD(3)
FD(3)=D1*F(4,I)+D2*HF(I,4)
147 CALL TAYLOR(DETA(I-1),FD(2),FD(1),ZM(1,I-1))
C SAVE HISTORIC VALUES
162 DO 164 I=1,NETA
HF(I,4)=F(4,I)
HF(I,5)=D1*F(1,I)+D2*HF(I,1)
DO 164 J=1,3
HF(I,J)=F(J,I)
164 CONTINUE
C COMPUTE GROUPINGS WHICH DEPEND ON DZ
130 C1=1.+D7
BETA=BETAM(IS)
C2=-2.*(C1+BETA+DZ)

```

```

      C3=C3M(IS)
      C4=RHOM*WU*C3*C3
C     COMPUTE TWO OR THREE POINT DIFFERENCE TIME DERIVATIVES
      IF(KR(4)-1)200,200,201
200  IF(ITFM-2)202,203,204
201  IF(ITFM-2)202,203,203
202  T7=0.
      T1=0.
      T2=0.
      GO TO 205
203  T7=1./DELT
      T1=-T7
      T2=0.
      GO TO 205
204  T7=1.5/DELT
      T1=-2./DELT
      T2=.5/DELT
C     SAVE HISTORIC VALUES.
205  DO 206 I=1,NETA
      TF(I)=T1*FTM1(1,I)+T2*FTM2(1,I)
206  CONTINUE
207  CONTINUE
9904  FORMAT(4X I2,6X I2/(8X1P10F10.3))
      IF(KR(17)) 9905,9906,9905
9905  CONTINUE
      WRITE(KCUT,9907)
9907  FORMAT(2X24HDEBUG IS,DLX1 ... 7M,4F,TF)
      NETAM1=NETA-1
      WRITE(KCUT,9904) IS,ITEM,DLX1,DLX2,D7,D1,D2,TZ,T1,T2,
      IC1,C2,C3,C4,FD,((ZM(I,J),J=1,NETAM1),I=1,4),((HF(I,J),
      2J=1,5),I=1,NETA)
9906  CONTINUE
      RETURN
      END

```

```

      SUBROUTINE TAYLOR (D,FM,F,P)
      DIMENSION FM(1),F(1),P(1)
      COMMON/INTCON/KR(20),KIN,KCUT,MAT1I,MAT1J,NETA,I,IS,NS
1  IT,NTIME,NAM,NLEQ,NNLEQ,NNPL,ITS,CASE(15),B(8),MWE,NO
2  N,ITEM,NITEM,KR17,NBT,NBT2,IDENT,UMES,FREQ,EPS
      D2=D*D
      IF (KR(10)) 1,1,2
2  IF (I-NETA) 3,1,4
4  FD=0.
      P(1)=(((FM(3)/A.-FD/24.)*D-F(2)/2.)*D+F(1))*D
      P(2)=(((FD/30.-FM(3)/R.)*D+F(2)/3.)*D-F(1)/2.)*D2
      P(3)=0.
      P(4)=(((FM/20.-FD/72.)*D-F(2)/R.)*D+F(1)/6.)*D2*D-P(3)
1(3)
      GO TO 3
1  FD=F(3)-FM(3)
      P(1)=(((F(3)/6.-FD/24.)*D-F(2)/2.)*D+F(1))*D
      P(2)=(((FD/30.-F(3)/R.)*D+F(2)/3.)*D-F(1)/2.)*D2
      P(4)=(((FD/252.-F(3)/72.)*D+F(2)/30.)*D-F(1)/24.)*D2*
1D
      P(3)=(((F(3)/20.-FD/72.)*D-F(2)/R.)*D+F(1)/6.)*D2*D-P(
14)
3  CONTINUE
      RETURN
      END

```

```

      SUBROUTINE ITERAT
      COMMON/BJUMP/BIJMP,CORMA,EASE,ICORM,WDOT,I777,ISP,IX
      COMMON/ERRCON/FLF(73),FLA(73),FLEM,ELM(3),ELMM,IPLM,NE
1  LM,IPLMM,DFL(73),FNLF(28),ENL(28),FNLEM,FNLN(3),ENLMM,I
2  ENLM,NENLM,IPLMM,DFNL(28),DRNL(6)
      COMMON/INTCON/KR(20),KIN,KCUT,MAT1I,MAT1J,NETA,I,IS,NS
1  IT,NTIME,NAM,NLEQ,NNLEQ,NNPL,ITS,CASE(15),B(8),MWE,NO
2  N,ITEM,NITEM,KR17,NBT,NBT2,IDENT,UMES,FREQ,EPS
      COMMON/VARCON/F(5,25),FTM1(5,25),FTM2(5,25)

```

```

1 FORMAT(22H NON-CONVERGENT OUTPUT /)
5 FORMAT(6X I3, 1X F9.3, 2(1X F9.4), 2(2X I3, 2X 1P F10.3))
7 FORMAT(/10X 62H ITERATED VALUES      DAMP      MAX LI
1 INEAR      MAX MOMENTUM /6X 62HITS      TIME      FPPW
2          ERROR      ERROR)
181 CLOCKB=0.
    ITS=ITS+1
    CALL LINCER
    NON=0
    CALL NONCER
    FPPW=F(3,1)
    IF(KR(4)+KR(17)+KR(19)+NON)189,189,192
189 IF(ITS-1)192,192,194
192 WRITE(KOUT,7)
194 CLOCK1=0.
    RTIME=CLOCK1-CLOCKB
    BTIME=.01*RTIME
    CLOCKB=CLOCK1
    WRITE(KOUT,5)ITS,RTIME,FPPW,EASE,IFLM,ELMM,IFNLM,FNLEM
1921 IF(ELMM+FNLM*.001/(EASE*RUMP)-.002)162,162,159
162 NON=0
    RETURN
159 IF(ITS-50) 161,160,160
160 WRITE(KOUT,1)
    WRITE(6,1)
    IF (FLMM+FNLM-.02) 162,162,1601
1601 NON=1
    RETURN
C   ITERATE OR OUTPUT
161 IF(KR(4)) 181,181,193
193 NON=-1
    RETURN
END

SUBROUTINE LINCER
COMMON/ERRCOM/FLF(73),ELA(73),FLEM,ELM(3),ELMM,IFLM,NE
1 LM,ILMM,DFL(73),FNLE(28),ENL(28),FNLEM,ENLM(3),ENLMM,I
2 FNLM,NENLM,INLMM,DFNL(28),DRNL(6)
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
192(24),LAP(100),BAL(73,28)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1 IT,NTIME,NAW,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2 N,ITEM,NITEM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/PRMCOM/TIME(50),PRE(50),PTET,S(50),REKAP(50),RN
10SF,VKAP,NDISC,INDISC(50),NSD(10),MSD(10),IPRE,RADNO,CO
2 NF,DELT,CYCLE
COMMON/ITEMCOM/SPDUM2(10),DER(50),DUMM1(25),SLOPE(25),R
1 EDUM(25),SDUM1(50),SDUM2(50),FWDUM(50),XICON(50),FWCO
2 N(50),DUDS(50)
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
C   EVALUATE LINEAR ERRORS FOR MOMENTUM AND ENERGY
NENLM=2
NENLM=NENLM
DO 401 I=2,NETA
    IF(KR(10)) 4000,4000,4001
    IF (I-NETA) 4003,4000,4003
4001 DUM1=B(4)
4000 DUM2=B(5)
    DUM3=B(2)
    DUM4=B(3)
    DUM5=B(1)
    DUM6=1.0
    GO TO 4002
4003 DUM1=B(3)
    DUM2=0.
    DUM3=B(1)
    DUM4=0.
    DUM5=1.0
    DUM6=0.
4002 FLF(I-1)=-F(1,I-1)+DETA(I-1)*F(2,I-1)+DSQ(I-1)*B(1)*F
1(3,I-1)+DCU(I-1)*(DUM1*F(4,I-1)+DUM2*F(4,I))-F(1,I)

```

```

M=I+NFTA-1
FLF(M)=- (F(2,I-1)+DETA(I-1)*F(3,I-1)+DSQ(I-1)*(DUM3*F(
14,I-1)+DUM4*F(J,I))-F(2,I))
M=I+2*NETA-2
401 FLF(M)=- (F(3,I-1)+DETA(I-1)*DUM5*(F(4,I-1)+DUM6*F(4,I)
1) - F(3,I))
FLF(NETA)=-F(2,1)
C DETERMINE MAXIMUM LINEAR ERRORS
404 CALL ARMAX(MAT11,FLF,ELMM,IFLM)
C FORM PRODUCT OF A**-1 AND LINEAR ERRORS
469 CALL MATONE(FLF)
473 CONTINUE
RETURN
END

```

```

SUBROUTINE ARMAX(N,X,XM,I)
DIMENSION X(1)
I=1
XM=ABS(X(1))
IF(N-1) 4,4,5
5 DO 3 J=2,N
XT=ABS(X(J))
IF(XM-XT) 2,3,3
2 XM=XT
I=J
3 CONTINUE
4 XM=X(I)
RETURN
END

```

```

SUBROUTINE NENCFR
REAL MU
DIMENSION CORAR(25)
COMMON/COMMON/BUUMP,CORMA,EASE,ICORM,WDOT,I777,ISD,IX
COMMON/COMMON/PEM(50),PTE(50),DUES,UF(50,50),PE(50,50)
1,2H2,MU,T,OSTP(50),IDSP
COMMON/COMMON/FLF(73),FL4(73),FLEM,FLM(3),FLMM,IFLM,NE
1LM,IFLM,DFL(73),FNLE(28),FNL(28),FNLEM,ENLM(3),ENLMM,I
2ENLM,FENLM,INLMM,DFNL(28),DRNL(6)
COMMON/COMMON/ETA(25),DETA(24),DSQ(24),DCU(24),BI(24),
1B2(24),LAR(100),BAI(73,28)
COMMON/COMMON/HTSCOM/T7,T1,T2,TE(25),C1,C2,C3,C4,BETA,ZM(5,25
1),XT(50),HF(25,5),HUF,HUUE,DLX2,C3M(50),RETAM(50)
COMMON/COMMON/KR(20),KIN,KOUT,MAT11,MAT1J,NETA,I,IS,NS
1,T,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),R(8),MWE,NO
2N,ITEM,NITEM,KR17,NRT,NB2,IDENT,IJMES,FRFQ,EPS
COMMON/COMMON/AM(28,28),OVNL(28),C5,C6,C7,C8,C9,C10,C1
11,C12
COMMON/COMMON/PMCOM/TIME(50),PRE(50),PTET,S(50),ROKAP(50),RN
10SE,VKAP,NDISC,INDISC(50),NSD(10),MSD(10),IPRE,RAPND,CO
2NE,DELT,CYCLE
COMMON/COMMON/PPCOM/PP,SC,XM(5)
COMMON/COMMON/SDUM1(50),SDUM2(50),SDUM3(50),SDUM4(50),SDUM5(50),SDUM6(50),SDUM7(50),SDUM8(50),SDUM9(50),SDUM10(50),SDUM11(50),SDUM12(50),SDUM13(50),SDUM14(50),SDUM15(50),SDUM16(50),SDUM17(50),SDUM18(50),SDUM19(50),SDUM20(50),SDUM21(50),SDUM22(50),SDUM23(50),SDUM24(50),SDUM25(50),SDUM26(50),SDUM27(50),SDUM28(50),SDUM29(50),SDUM30(50),SDUM31(50),SDUM32(50),SDUM33(50),SDUM34(50),SDUM35(50),SDUM36(50),SDUM37(50),SDUM38(50),SDUM39(50),SDUM40(50),SDUM41(50),SDUM42(50),SDUM43(50),SDUM44(50),SDUM45(50),SDUM46(50),SDUM47(50),SDUM48(50),SDUM49(50),SDUM50(50)
COMMON/COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
COMMON/COMMON/WALCOM/FW(50),RHQVW(50),IFW,IRHOVW,JRHOVW
EQUIVALENCE (CORAR(1),AM(1))
EASE=AMN1(EASE*2.,1.)
IF(ITS-1) 10,5,10
5 EASE=1.
BUUMP=1.0
4 ICORM=1
CORMA=1.5 + 10
10 CONTINUE
IX=0
EVALUATE COEFFICIENTS AND ERRORS FOR NONLINEAR
EQUATIONS.
INITIALIZE AM MATRIX
DO 15 I=1,NLEQ

```

CC

```

      FNLE(I)=0.
      DO 15 J=1,NNLEQ
15    AM(I,J)=C.
      EVAL: GROUPINGS WHICH CHANGE DURING ITERATION BUT ARE
      NOT F(ETA).
      IF (ITEM-2) 30,31,32
30    C5=0.
      GO TO 33
31    C5=TZ*ALOG(UE(IS,ITEM))+T1*ALOG(UE(IS,ITEM-1))
      GO TO 33
32    C5=TZ*ALOG(UE(IS,ITEM))+T1*ALOG(I))+T2*ALOG(UE(IS,ITEM-
      1(UE(IS,ITEM-2)))
33    CONTINUE
C----- EVAL COEFS, AM, AND ERRORS FOR EDGE BOUN CONDS
C      NEXT, EVALUATE NONLINEAR ERRORS
45    FNLE(NETA+1)=-F(1,-F(2,NETA))
      FNLE(NETA+2)=-F(3,NETA)
      FNLE(1)=FW(IS)-F(1,1)
C      NEXT, EVALUATE ORIGINAL AM COEFFICIENTS
      AM(1,1)=1.
      AM(NETA+2,2)=-1.
      FNLE(NETA+1)=FNLE(NETA+1)+FLE(2*NETA-1)
      DO 50 J=1,MAT1J
50    AM(NETA+1,J)=AM(NETA+1,J)+BA1(2*NETA-1,J)
C      START OF MAJOR DC LOOP FOR EVAL OF COEFFS AND ERRORS
C      AT EACH STATION.
      DO 120 I=1,NETA
C      TEST TO BYPASS COMMANDS THAT CANNOT BE PERFORMED AT
C      ETA(1).
      IF (I-1) 75,75,55
55    CALL IMONE
C----- EVAL GROUPINGS WHICH ARE USED AT I-1 AS WELL AS AT I
75    C6=HF(I,5)
      C7=F(3,1)+F(2,I)*(C1*F(1,I)+C6)
      C8=C4*(TZ-C5)
      C9=C1*F(2,I)-C8
      C10=C6+C1*F(1,I)
      C11=-C4*F(1,I)
      C12=C9*F(1,I)
      IF (I-1) 100,105,100
C      BACK TO CONSERVATION EQUATIONS
100   CALL IONLY
105   IF (KR(17)) 120,120,115
110   FORMAT(21H ALL THE COEFFICIENTS/(1X1P12E10.3))
115   WRITE(KOUT,110)C1,C2,C3,C4,C5,C6,C7,C8,C9,C10,C11,C12
      IX=-2
120   CONTINUE
145   IF (KR(19)) 170,190,170
170   CONTINUE
      WRITE(KOUT,175)
175   FORMAT(2X10HDEBUG FNLE )
180   FORMAT(/(12X1P10E10.3))
      WRITE(KOUT,180)(FNLE(J),J=1,MAT1I)
190   CONTINUE
      DO 195 III=1,MAT1J
195   ENL(III)=FNLE(III)
      CALL RERAY(MAT1J,AM,FNL,1,IX,2P)
      IF (KR(17)) 245,265,245
245   CONTINUE
      WRITE(KOUT,255)
255   FORMAT(2X9HDEBUG FLE )
      WRITE(KOUT,180)(FLE(J),J=1,MAT1I)
265   CONTINUE
C      DETERMINE MAXIMUM NONLINEAR ERRORS
      CALL ABMAX(MAT1J,FNLE,FNLEM,IFNLM)
      ELMM = ABS(ELMM)
      ENLMM=ABS(FNLEM)
C      EVALUATE NONLINEAR CORRECTIONS
      DO 615 I=1,MAT1J
615   DVNL(I)=ENL(I)
625   CONTINUE
C      EVALUATE LINEAR CORRECTIONS

```

```

      DO 630 I=1,MAT1I
      DO 630 J=1,MAT1J
630  FLE(I) = FLE(I) - DVNL(J) * BAL(I,J)
      FLE(NETA)=0.
      DO 620 I=1,NETA
      L=I+NETA-1
620  CORAR(I)=FLE(L)
655  CONTINUE
      IF (.33 + CORAR(ICORM) / CORMA) 670,675,675
670  IF (EASE-.2) 675,671,671
671  BUMP = 2.*BUMP
675  CALL ARMAX(NETA,CORAR,CORMA,ICORM)
      IF (KR(17)) 680,680,685
680  IF (KR(19)) 690,705,690
685  CONTINUE
      KR(17) = KR(17) - 1
690  CONTINUE
695  FORMAT(2X23HDERUG CORRECTIONS FL,NL )
      WRITE(KCUT,695)
      WRITE(KCUT,180) (FLE(J),J=1,MAT1I)
      WRITE(KCUT,180) (DVNL(J),J=1,MAT1J)
705  CONTINUE
C      CORRECT PRIMARY VARIABLES
      DUM=.1/RUMP
      EASE=AMIN1(3.*EASE,1.,DUM/ABS(CORMA))
710  IF (KR(13)) 720,720,715
715  DUM = KR(13)
      EASE = AMIN1(DUM / 10.,EASE)
720  IF (EASE - 1.0) 725,740,740
725  DO 730 I=1,MAT1I
730  FLE(I) = FLE(I) * EASE
      DO 735 I=1,MAT1J
735  DVNL(I) = DVNL(I) * EASE
740  CONTINUE
750  DO 765 J=2,NETA
      F(1,J) = F(1,J) + FLE(J - 1)
765  CONTINUE
      M = NETA
      DO 780 J=1,NETA
      M = M + 1
      F(2,J) = F(2,J) + FLE(M - 1)
780  CONTINUE
      DO 785 J=2,NETA
      M = M + 1
785  F(3,J - 1) = F(3,J - 1) + FLE(M - 1)
800  F(1,1)=F(1,1)+DVNL(1)
      F(3,NETA)=F(3,NETA)+DVNL(2)
      DO 835 J=1,NETA
      F(4,J)=F(4,J)+DVNL(J+2)
835  CONTINUE
      IF (ITS - 49) 850,840,850
840  IF (1777 - 777) 845,850,845
845  1777 = 777
      ITS = 30
850  CONTINUE
      RETURN
      END

```

```

SUBROUTINE IMONE
DIMENSION RRP(5)
COMMON/ERRCON/FLE(73),FLA(73),FLEM,ELM(3),ELMM,IFLM,NE
1LM,ILMM,DFL(73),FNLE(28),ENL(28),FNLEM,ENLM(3),ENLMM,I
2FNLM,NFLM,INLMM,DENL(28),DNL(5)
COMMON/FTACOM/ETA(25),DETA(24),DSQ(24),DCU(24),BI(24),
1R2(24),LAR(100),BAL(73,28)
COMMON/HTSCOM/TZ,T1,T2,TF(25),C1,C2,C3,C4,BETA,ZM(5,25
1),XT(50),HF(25,5),HUF,HHUF,DLX2,C3M(50),RETAM(50)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAM,NLEQ,NLLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NITEM,KR17,NRT,NBT2,IDENT,UMFS,FRFQ,EPS
COMMON/NONCOM/AM(28,28),DVNL(28),C5,C6,C7,C8,C9,C10,C1

```

```

11,C12
COMMON/PRPCOM/PR,SC,XM(5)
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
C EVALUATE GROUPINGS WHICH CONTRIBUTE TO (I-1) PORTION
C OF COEFFS VARIABLES WITH DIMENSION (NETA-1)
IF(I-2)401,4002,401
4002 XM(5)=C
C EVALUATE XM, WHICH CONTRIBUTES TO ERRORS AND TO COEFFS
C AT (I) AND (I-1)
401 CALL TAYLOR (DETA(I-1),F(2,I-1),F(2,I),XM)
C EVAL PORTION OF NLE DEPENDENT ON XM,... AND GROUPINGS
C EVAL AT (I-1).
403 C72=F(2,I)*XM(1)+F(3,I)*XM(2)+F(4,I)*XM(3)+F(4,I-1)*XM
1(4)
XM(5)=XM(5)+C72
C73=F(2,I)*7M(1,I-1)+F(3,I)*ZM(2,I-1)+F(4,I)*ZM(3,I-1)
1+F(4,I-1)*7M(4,I-1)
FNLE(I)=-C7-C11+DETA(I-1)/2.*(BETA+C4*C5)+.5*C2*C72-
12.*C73)
C RPP(1)=FNLE(I)
C EVAL PORTION OF ORIG COEFFS OF AM DEPENDENT UPON PARAM
C EVAL AT (I-1).
405 AM(1,I+1)=C2*XM(4)-2.*ZM(4,I-1)
407 IF(I-2) 418,417,418
417 AM(2,I)=-C6
C EVAL I-1 PORTION OF COEFFS, AND FORM CONTRIBUTIONS TO
C AM AND ERRORS.
418 LPI=NETA-2+I
DUM1=-C10
421 CALL AMSET (DUM1,LPI,2,RRR)
411 LPI=LPI+NETA
DUM1=-1.
423 CALL AMSET (DUM1,LPI,3,RRR)
IF(I-2)424,425,424
424 CALL AMSET (-C9,I-2,4,RRR)
425 CONTINUE
RRR(5)=FNLE(I)
9999 FORMAT(2X,I5,2X1D5E12,4)
IF(KR(17)) 9901,9902,9901
9901 CONTINUE
WRITE(KOUT,9001)
9001 FORMAT(34H DEBUG FNLE FROM IMONE PLUS COEFS )
WRITE(KOUT,9999)I,RRR
9902 CONTINUE
RETURN
END

```

```

SUBROUTINE AMSET (CAM,LPI,NN,RRR)
DIMENSION RRR(5)
COMMON/ERRCOM/FLF(73),ELA(73),FLEM,ELM(3),ELMM,IFLM,NE
1LM,ILMM,DFL(73),FNLE(28),ENL(28),FNLEM,ENLM(3),ENLMM,I
2ENLM,NENLM,ILMM,DFNL(28),DRNL(6)
COMMON/ETACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
1B2(24),LAR(100),BA1(73,29)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,I,T,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NITEM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,FPS
COMMON/NCNCOM/AM(28,28),DVNL(28),C5,C6,C7,C8,C9,C10,
1C11,C12
FNLE(I)=FNLE(I)-CAM*FLF(LPI)
IF(NN-1)2,1,1
1 RRR(NN)=FNLE(I)
2 DO 3 J=1,MAT1J
3 AM(I,J)=AM(I,J)-CAM*BA1(LPI,J)
RETURN
END

```

```

SUBROUTINE ICNLY
DIMENSION RRR(4)
COMMON/ERRCOM/FLF(73),ELA(73),FLEM,ELM(3),ELMM,IFLM,NE

```

```

1LM,ILMM,FEL(73),FNLE(28),ENL(28),FNLEM,ENLM(3),ENLMM,I
2FNLFM,NEFLM,INLMM,FENL(28),DRNL(6)
COMMON/FTACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
182(24),LAR(100),BA1(73,28)
COMMON/HISCOM/T7,T1,T2,TF(25),C1,C2,C3,C4,BETA,ZM(5,25
1),XI(50),HF(25,5),HUE,HHUE,DLX2,C3M(50),PETAM(50)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NTITEM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/NONCOM/AM(28,28),DVNL(28),C5,C6,C7,C8,C9,C10,
1C11,C12
COMMON/PPPCOM/PR,SC,XM(5)
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
C ADD CONTRIBUTIONS OF I TO NONLINEAR ERRORS
C EVALUATE GROUPINGS WHICH ARE USED ONLY AT I (NOT AT
C I-1).
  FNLE(I)=FNLE(I)-(C7+DETA(I-1)/2.*(BETA+C4*C5)+C11)
  RRR(1)=FNLE(I)
C EVAL PORTION OF ORIG COEFFS OF AM DEPENDENT UPON PARAM
C EVAL AT I.
403 AM(I,I+2)=AM(I,I+2)+C2*XM(3)-2.*ZM(3,I-1)
C ADD CONTRIBUTIONS OF I DEPENDENT COEFFS TO AM AND
C NONLINEAR ERRORS.
  IF(I-NETA)438,438,438
438 AM(I,2)=AM(I,2)+1.+C2*XM(2)-2.*ZM(2,I-1)
438 CALL AMSET(C9,I-1,2,RRR)
  LPT=NETA-1+I
  DUM1=C11+C2*XM(1)-2.*ZM(1,I-1)
444 CALL AMSET(DUM1,LPI,3,RRR)
418 IF(I-NETA)455,455,455
455 LPT=2*NETA-1+I
  DUM1=1.+C2*XM(2)-2.*ZM(2,I-1)
461 CALL AMSET(DUM1,LPI,4,RRR)
405 CONTINUE
9900 FORMAT(2X,I5,2X1P4E12.4)
  IF(KR(17)) 9901,9902,9901
9901 CONTINUE
  WRITE(KOUT,9901)
9901 FORMAT(34H, DEBUG FNLE FROM IONLY PLUS COEFS )
  WRITE(KOUT,9999) I,RRR
9902 CONTINUE
  RETURN
END

```

```

SUBROUTINE OUTPUT
PEAL MU
DIMENSION CM(5),U(25),Y(25)
COMMON/EDGCOM/DEM(50),PTE(50),DUES,UE(50,50),PE(50,50)
1,RHO,MU,T,DSIP(50),IDSIP
COMMON/FTACOM/ETA(25),DETA(24),DSQ(24),DCU(24),B1(24),
182(24),LAR(100),BA1(73,28)
COMMON/HISCOM/T7,T1,T2,TF(25),C1,C2,C3,C4,BETA,ZM(5,25
1),XI(50),HF(25,5),HUE,HHUE,DLX2,C3M(50),PETAM(50)
COMMON/INTCOM/KR(20),KIN,KOUT,MAT1I,MAT1J,NETA,I,IS,NS
1,IT,NTIME,NAM,NLEQ,NNLEQ,NRNL,ITS,CASE(15),B(8),MWE,NO
2N,ITEM,NTITEM,KR17,NBT,NBT2,IDENT,UMFS,FREQ,EPS
COMMON/NONCOM/AM(28,28),DVNL(28),C5,C6,C7,C8,C9,C10,
1C11,C12
COMMON/PRPCOM/PR,SC,XM(5)
COMMON/TERCOM/SDUM2(10),DFR(50),DUMM1(25),SLOPE(25),R
1SDUM(25),SDUM1(50),SDUM2(50),FWDUM(50),XICON(50),FWCO
2N(50),DURS(50)
COMMON/VARCOM/F(5,25),FTM1(5,25),FTM2(5,25)
COMMON/WALCOM/FW(50),RHOV(50),TFW,IRHOV,IRHOV
1 FORMAT(/9X65HEDGE STATIC XI
1 FLUX NORM ROKAP /6X57HVELOCITY PRESSURE
2(LB/ RFTA PARAMETER /6X57H(FT/SEC)
3(ATM) SEC)**2 (FT)

```



```

308 DUM3=RHO*UE( IS,ITEM)
   DUM4=-1./C3
309 CONTINUE
   IF(KR(14)) 310,311,310
310 WRITE(KOUT,6)((F(I,J),J=1,NETA),I=1,4)
311 RETURN
   END

```

LIST OF REFERENCES

1. Schlichting, H., Boundary Layer Theory, 4th Ed., p. 107-265, McGraw-Hill, 1960.
2. Lighthill, M. J., "Response of Laminar Skin Friction and Heat Transfer to Fluctuations in Stream Velocity," Proceedings of the Royal Society of London, A 224, June 1954.
3. Glauert, M. B., "The Laminar Boundary Layer on Oscillating Plates and Cylinders," Journal of Fluid Mechanics, v. 1, Part 1, May 1956.
4. Carrier, G. F. and DiPrima, R. C., "On the Unsteady Motion of a Viscous Fluid Past a Semi-Infinite Flat Plate," Journal of Mathematics and Physics, January 1957.
5. Nickerson, R. J., The Effect of Freestream Oscillations on the Laminar Boundary Layer on a Flat Plate, D.Sc. Thesis, Massachusetts Institute of Technology, 1957.
6. Hill, P. G. and Stenning, A. H., "Laminar Boundary Layers in Oscillatory Flow," Transactions of the ASME: Journal of Basic Engineering, A 265, September 1960.
7. Koob, S. J. and Abbott, D. E., "Investigation of a Method for the General Analysis of Time Dependent Two-Dimensional Laminar Boundary Layers," Transactions of the ASME: Journal of Basic Engineering, NO. 4, 1968.
8. Los Alamos Scientific Laboratory Report LA-2910, A Method for Computing Nonsteady, Incompressible, Viscous Fluid Flows, by J. E. Fromm, May 1963.
9. Chorin, A. J., "A Numerical Method for Solving Incompressible Viscous Flow Problems," Journal of Computational Physics, v. 2, 1967.
10. Smith, A. M. O. and Clutter, D. W., "Solution of the Incompressible Laminar Boundary Layer Equations," AIAA Journal, v. 1, No. 9, September 1963.
11. Aerospace Research Laboratories Report ARL 67-0050, Finite Difference Solution of the Laminar Boundary Layer Equations in the Von Mises Variables with Applications, by G. Kleinstein, March 1967.

12. Aerospace Research Laboratories Report ARL 67-0098, Operation Instructions and Description of Computer Program for Finite Difference Solution of the Laminar Compressible Boundary Layer Equations in the Von Mises Variables, by F. Nabi, May 1967.
13. Aerospace Research Laboratories Report ARL 67-0045, Solution of the Laminar Boundary Layer Equations with Mass Transfer by the GKD Multimoment Method, by H. E. Bethel, March 1967.
14. Aerospace Research Laboratories Report ARL 67-0052, An Approximate Solution of Von Mises' Boundary Layer Equation, by A. Reggiori, March 1967.
15. Aerospace Research Laboratories Report ARL 67-0124, On an Alternative Method of Numerically Evaluating the Linearized Problem of an Impulsively Moving Flat Plate in a Rarefied Gas Flow, by K. S. Nagaraja and J. P. Hudson, June 1967.
16. Aerotherm Corporation Report No. 66-7, Part III, Nonsimilar Solution of the Multicomponent Laminar Boundary Layer by an Integral Matrix Method, by E. P. Bartlett and R. M. Kendall, NASA CR-1062, June 1968.
17. Howarth, L., "On the Solution of the Laminar Boundary Layer Equations," Proceedings of the Royal Society of London, A 164, January-February 1938.
18. Aerotherm Corporation Report No. 68-42, User's Manual Boundary Layer Integral Matrix Procedure (BLIMP), by E. P. Bartlett and C. Deblaye, October 1968.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2. Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3. Commander, Naval Air Systems Command Department of the Navy Washington, D.C. 20360	2
4. Dean of Research Administration Naval Postgraduate School Monterey, California 93940	2
5. Chairman, Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1
6. Professor James A. Miller Department of Aeronautics Naval Postgraduate School Monterey, California 93940	2
7. LT Barry A. Gastrock, USN 34 Reeceville Road Coatesville, Pennsylvania 19320	2

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)		2a. REPORT SECURITY CLASSIFICATION	
Naval Postgraduate School Monterey, California 93940		Unclassified	
		2b. GROUP	
3. REPORT TITLE			
NONSIMILAR SOLUTION OF THE LAMINAR BOUNDARY LAYER IN AN OSCILLATORY FLOW BY AN INTEGRAL MATRIX METHOD			
4. DESCRIPTIVE NOTES (Type of report and, inclusive dates)			
Aeronautical Engineer's Thesis, June 1969			
5. AUTHOR(S) (First name, middle initial, last name)			
Barry Allen Gastrock, Lieutenant, United States Navy			
6. REPORT DATE		7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1969		98	18
8a. CONTRACT OR GRANT NO.		9a. ORIGINATOR'S REPORT NUMBER(S)	
b. PROJECT NO.			
c.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.			
10. DISTRIBUTION STATEMENT			
Distribution of this document is unlimited.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY	
		Naval Postgraduate School Monterey, California 93940	
13. ABSTRACT			
The development of a numerical procedure for the treatment of nonsimilar, unsteady, laminar boundary layers is presented. The solution is obtained from the laminar, isothermal, incompressible boundary layer equations employing a modification of the integral matrix procedure of Bartlett and Kendall. Solutions of example problems are presented for steady Blasius and Howarth flows, and for oscillating Blasius flow. Agreement with the known classical results is satisfactory and establishes the general feasibility of the method. Core storage requirements of 130,000 bytes allow consideration of as many as 25 nodal points across the boundary layer, 50 time increments per oscillation cycle and 50 streamwise stations. Solution of oscillating Blasius flow considering 8 nodal points and 16 time increments requires 13.49 seconds for one streamwise station utilizing IBM 360/67 time sharing capabilities.			

UNCLASSIFIED

Security Classification

14

KEY WORDS

LINK A

LINK B

LINK C

ROLE

WT

ROLE

WT

ROLE

WT

BOUNDARY LAYER

LAMINAR BOUNDARY LAYERS

BOUNDARY LAYER SOLUTION

BOUNDARY LAYER COMPUTER PROGRAM

UNSTEADY BOUNDARY LAYER





thesG2357

Nonsimilar solution of the laminar bound



3 2768 002 01087 8

DUDLEY KNOX LIBRARY